AIWolfDial 2019

Proceedings of the 1st International Workshop of AI Werewolf and Dialog System

Proceedings of the Workshop

October 29, 2019
Tokyo, Japan

Order copies of this and other ACL proceedings from:

Association for Computational Linguistics (ACL)
209 N. Eighth Street Stroudsburg, PA 18360
USA
Tel: +1-570-476-8006
Fax: +1-570-476-0860
acl@aclweb.org

ISBN 978-1-948087-91-9

Introduction

This volume contains the papers presented at W19-81 1st International Workshop of AI Werewolf and Dialog System (AIWolfDial) held on October 29, 2019 in conjunction with INLG 2019 in Tokyo.

“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.

The aim of the workshop was to bring together researchers interested in techniques for dialogue systems and game AIs, not limited for the werewolf game. This includes studies of conversation game AIs, mental models, agent systems, negotiation strategies, etc.

Yoshinobu Kano, Claus Aranha, Michimasa Inaba, Fujio Toriumi, Hirotaka Osawa, Daisuke Katagami, Takashi Otsuki
October 2019

Workshop Organizers:

Yoshinobu Kano, Shizuoka University, Japan
Claus Aranha, Tsukuba University, Japan
Michimasa Inaba, The University of Electro-Communications, Japan
Fujio Toriumi, The University of Tokyo, Japan
Hirotaka Osawa, University of Tsukuba, Japan
Daisuke Katagami, Tokyo Polytechnic University, Japan
Takashi Otsuki, Yamagata University, Japan

Program Committee:

Yoshinobu Kano, Shizuoka University, Japan
Claus Aranha, Tsukuba University, Japan
Michimasa Inaba, The University of Electro-Communications, Japan
Fujio Toriumi, The University of Tokyo, Japan
Hirotaka Osawa, University of Tsukuba, Japan
Daisuke Katagami, Tokyo Polytechnic University, Japan
Takashi Otsuki, Yamagata University, Japan
Hitoshi Matsubara, Future University Hakodate, Japan

Invited Speaker:

Yoshinobu Kano, Shizuoka University, Japan

Table of Contents

Workshop Program

Tuesday, October 29, 2019

10:00-10:05	Opening
10:05-10:15	Overview
10:15-10:55	Invited talk Hirotaka Osawa
10:55-11:15	*Data Augmentation Based on Distributed Expressions in Text Classification Tasks* Yu Sugawara
11:15-11:35	*Are Talkative AI Agents More Likely to Win the Werewolf Game?* Dolça Tellols
11:35-11:55	Coffee Break
11:55-12:15	*AI Werewolf Agent with Reasoning Using Role Patterns and Heuristics* Issei Tsunoda, Yoshinobu Kano
12:15-12:35	*Strategies for an Autonomous Agent Playing the "Werewolf game" as a Stealth Werewolf* Shoji Nagayama, Jotaro Abe, Kosuke Oya, Kotaro Sakamoto, Hideyuki Shibuki, Tatsunori Mori, Noriko Kando
12:35-12:45	Shared task awards
12:45-13:00	Demonstration
13:00-13:10	Discussion
13:10-13:15	Closing

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 2019

Data Augmentation Based on Distributed Expressions in Text Classification Tasks

Yu Sugawara
Faculty of Information Science and Technology, Hokkaido University
suga@ist.hokudai.ac.jp

Abstract

We propose a data augmentation method that combines Doc2vec and Label spreading in text classification tasks. The feature of our approach is the use of unlabeled samples, which are easier to obtain than labeled samples. We use them as an aid to the classification model to improve the accuracy of its prediction. We used this method to classify several text data sets including the natural language branch of the AIWolf contest. As a result of the experiments, we confirmed that the prediction accuracy is improved by applying our proposed method.

1 Introduction

Analyzing human intentions in texts is a task in high demand in natural language processing. On the other hand, to solve this task well, it is necessary to prepare an enormous amount of natural language corpora that the intentions of each text are labeled. In particular, if the context is unusual, like in-game conversations, the preprocessed training data that meets the demand is rarely available. Thus we have to manually label intentions one by one or pay for crowdsourcing.

To cope with this situation, we propose a method that can estimate the intention of texts with high accuracy from a large number of unlabeled samples and a relatively small amount of labeled ones.

1.1 Data augmentation via unlabeled samples

There are several existing methods for performing data augmentation based on unlabeled samples. In S-EM(Nigam et al., 2000), a naive Bayes model is first constructed using only labeled samples. The trained naive Bayes model gives unlabeled samples an estimated probability of their label. Then, a new naive Bayes model is constructed using all the samples, both originally labeled and newly labeled. As with the EM algorithm, this procedure is repeated until the parameters of the model converge.

Many of the related methods involve minor changes to S-EM, such as replacing the algorithm used in intermediate steps with a more accurate one(Li and Liu, 2003).

1.2 Word2vec and Doc2vec

Word2vec(Mikolov et al., 2013) is a method that expresses a word as a distributed representation with a high dimensional vector. The regularity of addition and subtraction is shown by vector representation of words such that vector('king') - vector('man') + vector ('woman') approximates vector ('queen'). Word2Vec uses a Bag-of-Words model, which uses the number of occurrences of words in a sentence, and a Skip-gram model, which uses the word occurrence probability from the sequence of words in a sentence.

Doc2vec(Le and Mikolov, 2014) is a method to perform the same operation as Word2vec on a document. It converts a document into a vector representation in high-dimensional space. As with Word2vec, documents that are close in this space can be interpreted as having a similar context.

1.3 Label spreading

Label spreading(Zhou et al., 2003) is a semi-supervised learning method. The goal of semi-supervised learning is to estimate the label of unlabeled samples based on a small number of labeled samples. In label spreading, the label information is propagated from the labeled sample to the unlabeled sample at a close distance. This newly labeled sample also has a influence on the surrounding sample. By repeating this propagation, the label information of labeled samples is spread for all samples.

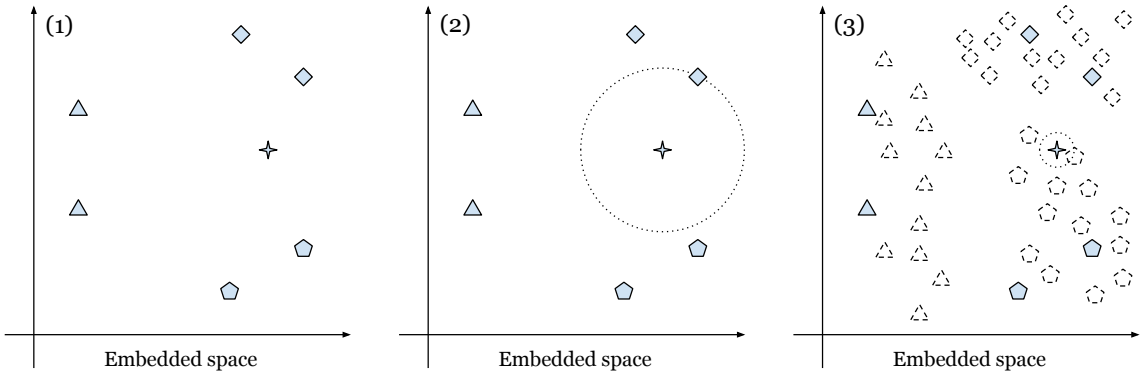

Figure 1: Concept of the proposed method. The figures enclosed by solid lines represent labeled samples embedded in the space. Those enclosed by dashed lines represent originally unlabeled samples. The difference in shape represents the label that the sample has.

2 Our proposed method

We propose a method to estimate the true label of the documents with high accuracy but from a relatively small amount of labeled data.

The model training process is as follows. First, we perform a word segmentation via morphological analysis on all the documents to obtain an ordered list of words. This operation is peculiar to the Japanese language, which is not normally written with a space between words. For that, it may not be necessary when applying this method to other languages such as English. Based on the result, the Doc2vec model is constructed using both labeled and unlabeled training samples. Thus each sample is made to correspond to the coordinate of the high dimensional space. After that, Label spreading is performed in this space. Labeled samples are used to label all the remaining unlabeled samples. The label information is propagated to surrounding samples in embedding space.

In the prediction process, we input the natural language document to the previously trained Doc2vec model to get the vector representation of the sample in the high dimensional space. The Nearest centroid algorithm(Tibshirani et al., 2002) is performed in this space, which estimates the label of the sample based on the neighboring samples. Finally, the true label of this sample is estimated.

We show this method schematically in Figure 1. (1) Our objective is to estimate the label of the sample embedded in the star position. (2) If we simply apply the nearest neighbor algorithm by using just labeled samples, the estimation is not reliable. (3) In our proposed method, the label of unlabeled samples is complemented by Label spreading at first. The Nearest centroid algorithm is applied based on both originally and newly labeled samples.

3 Experiments

3.1 Experimental setting

To verify the effectiveness of the proposed method, we conducted the following experiments. First, we prepared corpora that the intentions are labeled on. Then, we remove the label information from about 90% of the datasets. We trained the Doc2vec model with both labeled and unlabeled data, then use it to embed all samples to high dimensional space. After that, we performed Label spreading to recover label information. For comparison, we also prepared a model that simply executes the Nearest centroid using only the labeled data. Finally, we input the corpora not used for training and compared the prediction accuracy of the true label.

For Label spreading and Nearest centroid, we used the implementations of scikit-learn(Pedregosa et al., 2011).

3.2 Datasets

The following three corpora were used in this experiment.

Livedoor consists of documents published in an online news site. We labeled the topic category in which the news appears. There are nine categories such as ”sports”, ”life hacks” and so on. Our purpose is to estimate the topic category from a news article.

Label	Example (Japanese)
ASK_WHO_LIKE_WEREWOLF	>>Agent[xx] 君は誰が怪しいと考えているのかな？
ASK_WHY_DIVINE	>>Agent[xx] Agent[xx] はどうして私を占ったんだい？
COMINGOUT_VILLAGER	私は人間さ。
COMINGOUT_WEREWOLF	実は私が狼だったんだよ。
DIVINED_HUMAN	占い CO。Agent[xx] は人間だったよ。
DIVINED_WEREWOLF	占い師は私だよ。昨日の結果だが、Agent[xx] は人狼だと出た。
ESTIMATE_HUMAN	Agent[xx] は人間だと思う。
ESTIMATE_WEREWOLF	Agent[xx] が怪しいと思っているよ。
UNIMPORTANT	おはよう。CO はあるかな？
REQUEST_VOTE	Agent[xx] に投票して欲しい。

Table 1: Labels we defined in the AIWolfNLP.

Table 2: Outline of the datasets used in the experiment. Each column indicates the number of labels, the number of unlabeled samples and the number of labeled samples.

	# labels	# unlabeled	# labeled
livedoor	9	6638	663
wolfBBS	9	9343	1038
AIWolfNLP	10	1653	212

WolfBBS consists of utterances generated by humans on Werewolf BBS, an online BBS for playing the Werewolf game. Nine intentions are defined such as "COMING OUT", "DIVINE RESULT", and so on. Each utterance is annotated one of nine intentions.

AIWolfNLP consists of the utterances in the natural language branch of the 4th AIWolf Contest. We labeled the intention of each utterance generated in the TALK phase. We defined 10 intentions that seem to be useful in understanding the game situation such as "DIVINED WEREWOLF", "REQUEST VOTE", and so on. Examples of the correspondence between each text and assigned label are shown in Table 1. Our purpose is to estimate the intention of the utterance. In this dataset, just one agent's utterances are labeled and others are unlabeled. This is a setting that assumes the case of actually participating in the natural language branch of the AIWolf contest. We have a complete set of utterances and intent pairs for the agents we created, but no information about other agents.

A summary of these datasets is presented in Table 2.

3.3 Experimental results

The experimental results for each dataset are shown in Table 3. In each dataset, the proposed method that exploits both labeled and unlabeled samples gained higher prediction accuracy than the method simply applying the Nearest centroid using just labeled samples.

Table 3: The prediction accuracy on validation samples. The simple method discards unlabeled samples and runs Nearest Centroid with only labeled data. The proposed method first completes the labels of unlabeled samples and then runs Nearest Centroid with all the data.

	simple	proposed
livedoor	71.7%	80.3%
wolfBBS	49.2%	50.7%
AIWolfNLP	15.8%	57.4%

4 Conclusion

We proposed an effective prediction method for document classification tasks when a large number of unlabeled samples and a few labeled samples are retained. Our experiments demonstrated that the proposed method gained significantly higher prediction accuracy than the model trained on only labeled samples. It is often the case that the text itself is available in large quantities, but only a few samples are labeled. This method will be quite useful in such situations.

As a prospect, we should conduct similar experiments on languages other than Japanese to confirm the usefulness of the method. The object of the experiment was limited to Japanese in this paper, but since this method has no language dependency, it can also be applied to any language.

References

Quoc V. Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *Pro-*

ceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014, pages 1188–1196.

Xiaoli Li and Bing Liu. 2003. Learning to classify texts using positive and unlabeled data. In *IJCAI-03, Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence, Acapulco, Mexico, August 9-15, 2003*, pages 587–594.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States.*, pages 3111–3119.

Kamal Nigam, Andrew McCallum, Sebastian Thrun, and Tom M. Mitchell. 2000. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103–134.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. 2011. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, 12:2825–2830.

R. Tibshirani, Trevor Hastie, B. Narasimhan, and Gilbert Chu. 2002. Diagnosis of multiple cancer types by shrunken centroids of gene expression. *Proceedings of National Academic Science (PNAS)*, 99:6567–6572.

Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. 2003. Learning with local and global consistency. In *Advances in Neural Information Processing Systems 16 [Neural Information Processing Systems, NIPS 2003, December 8-13, 2003, Vancouver and Whistler, British Columbia, Canada]*, pages 321–328.

Are Talkative AI Agents More Likely to Win the Werewolf Game?

Dolça Tellols
Tokyo Institute of Technology
tellols.d.aa@m.titech.ac.jp

Abstract

The Werewolf game is a communication game where, usually, two teams compete against each other. As players discuss and share ideas during the game to define their strategy, being talkative or not is one of the characteristics that define them. This paper presents a data analysis over logs from the shared task of The 1st International Workshop of AI Werewolf and Dialogue System to discuss if being talkative or not can be related to winning or losing when AI agents play the Werewolf game. Overall results show that the difference in the average of utterances sent by winning and losing players is not significant. However, they also suggest further analysis and discussion.

1 Introduction

In recent years, there have been approaches to implement Artificial Intelligence (AI) agents capable of playing and competing with other agents or human players in a variety of games. Proposals for games that do not require social interaction between the players, like chess, shogi and go, are achieving promising results (Silver et al., 2016, 2018). However, developing AI agents for communication games like the Werewolf game, where usually two teams (villagers and werewolves) discuss and compete against each other, remains a challenge.

The Artificial Intelligence based Werewolf project[1] is contributing to the previous aspect by researching and providing platforms to develop and test AI Werewolf agents. Researchers can implement agents to play in the protocol division (agents communicate in a middle language called the AI Werewolf protocol) or the natural language division (agents communicate using natural language utterances). The 1st International Workshop of AI Werewolf and Dialog System[2], taking place in the context of the International Natural Language Generation 2019 conference, proposed a shared task where participants implemented AI agents capable of playing the Werewolf game using natural language utterances (Kano et al., 2019).

In the frame of the shared task and based on the idea that being talkative or not may characterize the strategy that some players of the Werewolf game follow, this paper focuses on analyzing if the talkativeness level of the participant AI agents may be related to winning or losing the Werewolf game. To do so, and having as reference a talkative agent implemented to participate in the previously cited shared task, this work analyzes, in Section 4, the logs from the played games in which the agent participated and discusses the results in Section 5.

2 Related Work

In the last few years, research based on the Werewolf game is increasing.

On the one hand, a lot of researchers use the game to analyze human players' behaviors. For example, some created the Idiap Wolf Database and used it to show how it is possible to automatically detect suspicious actions and how the degree of speaker behavior influences on the outcomes of the game (Hung and Chittaranjan, 2010; Chittaranjan and Hung, 2010). Other researchers used machine learning to analyze video data of people playing the game and checked the importance that nonverbal information has to achieve victory (Katagami et al., 2014).

On the other hand, because of proposals like

[1] http://aiwolf.org/en/

[2] https://aiwolfdial.kanolab.net/home

the already cited AI based Werewolf project, the number of works defining implementation strategies for AI Werewolf agents is increasing. As an example, some researchers proposed psychological models to be used to implement AI Werewolf agents so that they achieve higher winning rates (Nakamura et al., 2016). And others developed a behavioral model for the implementation of agents based on logs from human players (Hirata et al., 2016).

This work contributes by providing a data analysis study that opens a discussion over how the talkativeness of an AI agent may be related or not to its winning rate. Results may serve as a reference for the future development of AI agents that can play the Werewolf game and communicate using natural language.

3 Dataset

As data, this paper uses a set of 60 game logs from the shared task of The 1st International Workshop of AI Werewolf and Dialogue System (AIWolfDial2019) from the 2019 International Natural Language Generation (INLG2019) conference (Kano et al., 2019).

In all games, the same five agents (A1, A2, A3, A4, and A5), play the werewolf game using natural language utterances written in Japanese. The talkative agent implemented for the shared task (A4) sends utterances as long as they do not become excessively repetitive (slight repetition may result in emphasis). All five agents participate in all games and, each time, they have randomly assigned one of the following roles: villager (has no special skill and there are two in each game), seer (can see if a player is human or werewolf at the end of each day), possessed (human from the werewolves side) or werewolf (can eliminate one player at the end of each day from day 1). Agents play each role 12 times (seer, possessed and werewolf cases) or 24 (villager case).

The shared task allows agents to communicate freely using natural language during certain periods ("days"), without specifying a maximum number of utterances per day. Since day 0 only consists of greetings, Section 4 analyzes utterances performed from day 1. Because of the small number of players, each game only lasts for one day (20 games) or two (40 games). This is because, each day from day 1, all alive players vote to eliminate a player (villagers try to use this voting to eliminate the werewolf) before the werewolf eliminates another one.

Logs contain the following information: (i) status (keeps playing or not) and role of each player at the beginning of each day and the end of the game; (ii) utterances each player performs during the day; (iii) information of the seer divination at the end of each day; (iv) voting each player performs at the end of each day (excluding day 0) and the corresponding result (which agent stops playing); (v) information of the werewolf attack at the end of each day (excluding day 0); and (vi) result of the game indicating the status and role of each player and the winning side (villagers or werewolves).

4 Data Analysis

To discuss in Section 5 if the talkativeness of AI agents affects their odds of winning the Werewolf game, this work analyzed the data presented in Section 3 from different points of view: (i) game result; (ii) side; (iii) role; and (iv) agent. For each case, this paper presents the average (Avg.) and standard deviation (Std. Dev.) of the number of utterances sent during one day by a player belonging to one of the categories each point of view may consider. When appropriate, it also presents data depending on the game result (win or lose) and shows the winning rate.

Agent	Utterances	
Game Result	Avg.	Std. Dev.
Win	8.9	1.05
Lose	9.08	0.88
All	8.99	0.97

Table 1: Analysis results of the utterances sent per agent and day according to the game result.

Table 1 shows an overview result by comparing the average of utterances sent each day by winning players and by losing players. Since the average number of utterances sent by winning players (8.9) is similar to the one of losing players (9.08), it seems that talkativeness may not be a determining factor that leads to deciding the game result. Because the difference between the winning and losing agents' data samples follows a normal distribution, this study also performed a t-test, which confirms that the difference in the presented averages is not significant (p-value = $0.3214 > 0.05$).

Agent Side	Result	Utterances Avg.	Utterances Std. Dev.	Win. Rate
Villagers	Win	9.22	0.68	0.62
	Lose	8.91	0.6	
	All	9.1	0.67	
Werewolves	Win	8.38	1.3	0.38
	Lose	9.18	1.01	
	All	8.88	1.19	

Table 2: Analysis results of the utterances sent per agent and day according to the side of the agent.

Table 2 shows the result of the analysis performed according to the side (villagers or werewolves) each agent belongs to in a game. On the one hand, winning players from the villagers' side perform more utterances (9.22) than losing players (8.91) and their winning rate is 0.62. On the other hand, losing players from the werewolves' side perform more utterances (9.18) than winning players (8.38) and their winning rate is 0.38. There is almost no difference between the average number of utterances performed by villagers' side players (9.1) and the average number of utterances performed by the werewolves' side players (8.88).

Agent Role	Result	Utterances Avg.	Utterances Std. Dev.	Win. Rate
Villager	Win	9.26	0.79	0.62
	Lose	8.96	0.93	
	All	9.14	0.86	
Seer	Win	9.15	1.17	0.62
	Lose	8.8	1.15	
	All	9.02	1.18	
Possessed	Win	8.15	1.71	0.38
	Lose	9.32	1.19	
	All	8.88	1.52	
Werewolf	Win	8.61	1.45	0.38
	Lose	9.04	1.37	
	All	8.88	1.42	

Table 3: Analysis results of the utterances sent per agent and day according to the role of the agent.

Table 3 illustrates the analysis result of the utterances sent by an agent according to its role. Results are coherent with the ones from Table 2, as possessed and werewolf role players (werewolves' side) tend to send more utterances when they lose while villager and seer role players (villagers' side) send more utterances when they win. In this table, we can also see how possessed and werewolf players, which have a lower winning rate, send fewer utterances on average than villager and seer players. Note that in the case of possessed players, there is an increase of 1.17 points on the average of utterances sent when they lose the game compared to the times when they win.

Agent	Result	Utterances Avg.	Utterances Std. Dev.	Win. Rate
A1	Win	7.63	1.08	0.52
	Lose	7.59	1.51	
	All	7.61	1.31	
A2	Win	9.63	0.77	0.52
	Lose	9.5	1.1	
	All	9.57	0.95	
A3	Win	9.54	0.46	0.38
	Lose	9.46	0.72	
	All	9.49	0.64	
A4	Win	9.57	0.63	0.58
	Lose	9.64	0.59	
	All	9.6	0.62	
A5	Win	8.64	1.65	0.62
	Lose	9.02	1.23	
	All	8.78	1.51	

Table 4: Analysis results of the utterances sent per agent in a day.

Finally, Table 4 presents the analysis results of the number of utterances sent per player in a day. As expected because of the talkativeness of A4, it presents the highest average of utterances sent from among all agents (9.6), which is 0.61 points above the average. Additionally, A4 also has the second-highest winning rate. It is interesting to observe though, how A5 has the highest winning rate but has the second-lowest average of sent utterances. Additionally, A4 and A5, the players with the highest winning rate, are the only ones with a higher average of utterances sent in losing games than in winning games.

5 Discussion

From the results obtained in Section 4, it seems that the number of utterances sent by the AI agents participating in the shared task was quite similar (around 9 utterances per player and day). The difference in the average of utterances sent by winning and losing players is also not significant. Consequently, we may be able to conclude that the number of utterances sent by the participant AI agents of the shared task may not be a significant

factor that determines the game result. However, since the winning rate of villagers' side players (0.62) is higher than the one of the werewolves' side players (0.38), it seems that other factors are leading certain players to victory.

One of the elements that may affect is the strategy implemented for each of the agents. As an example, the talkative agent implemented (A4) achieves a 0.46 winning rate when playing on the werewolves' side by following a strategy of faking the seer under certain circumstances.

Two other elements that may also affect are the content provided in an agent's utterances and the way it processes utterances performed by other agents. Note that, in the natural language division of the Werewolf game, performing appropriate utterances may be as important as listening and understanding the other agent utterances.

6 Conclusions and Future Work

This paper presented a data analysis on some logs from the shared task of AIWolfDial2019 workshop from INLG2019. The goal was to verify if the talkativeness of an AI agent could have an impact on the Werewolf game result.

Overall results showed that the number of utterances may not be a determinant factor influencing the result of the games played by the AI agents participating in the shared task.

Some questions that future work in this line may address could be: (i) are there some kinds of utterances that lead to winning or losing the game?; (ii) to what extent is the utterances' content important as far as a good game strategy is followed?; and (iii) how much do the other agents' strategy and conversational capabilities influence an agent's result?

This paper analyzed logs generated by only five AI werewolf agents, each with their own unique and independent strategies and conversational capabilities. Because the results of the analysis may depend on the implementation of the agents, it would also be interesting to analyze more data generated by a larger variety of agents.

References

Gokul Chittaranjan and Hayley Hung. 2010. Are you awerewolf? detecting deceptive roles and outcomes in a conversational role-playing game. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 5334–5337. IEEE.

Yuya Hirata, Michimasa Inaba, Kenichi Takahashi, Fujio Toriumi, Hirotaka Osawa, Daisuke Katagami, and Kousuke Shinoda. 2016. Werewolf game modeling using action probabilities based on play log analysis. In *International Conference on Computers and Games*, pages 103–114. Springer.

Hayley Hung and Gokul Chittaranjan. 2010. The idiap wolf corpus: exploring group behaviour in a competitive role-playing game. In *Proceedings of the 18th ACM international conference on Multimedia*, pages 879–882. ACM.

Yoshinobu Kano, Claus Aranha, Hirotaka Osawa, Daisuke Katagami, Takashi Otsuki, and Fujio Toriumi. 2019. Overview of the aiwolfdial 2019 shared task: Competition to automatically play the conversation game "mafia". In *In proceedings of the 1st International Workshop of AI Werewolf and Dialog System (AIWolfDial 2019), the 12th International Conference on Natural Language Generation (INLG 2019). Miraikan, Tokyo, Japan. 2019/10/29*.

Daisuke Katagami, Shono Takaku, Michimasa Inaba, Hirotaka Osawa, Kosuke Shinoda, Junji Nishino, and Fujio Toriumi. 2014. Investigation of the effects of nonverbal information on werewolf. In *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pages 982–987. IEEE.

Noritsugu Nakamura, Michimasa Inaba, Kenichi Takahashi, Fujio Toriumi, Hirotaka Osawa, Daisuke Katagami, and Kousuke Shinoda. 2016. Constructing a human-like agent for the werewolf game using a psychological model based multiple perspectives. In *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, pages 1–8. IEEE.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. 2016. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. 2018. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144.

AI Werewolf Agent with Reasoning Using Role Patterns and Heuristics

Issei Tsunoda **Yoshinobu Kano**
Faculty of Informatics, Shizuoka University, Hamamatsu, Shizuoka, Japan
itsunoda@kanolab.net, kano@inf.shizuoka.ac.jp

Abstract

The AIWolf project has been holding contests for these years to play the Werewolf game ("Mafia") by automatic agents. A difficulty of the Werewolf game is that the game is an imperfect information game, very small limited amount of information is shown to players, other than the player's own role information. Therefore, inference of probabilities for each player agent's role could not be confident theoretically, difficult to utter appropriate reasons when simply based on the probabilities. Focusing on a genuine seer and a fake seer, we implemented our 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.

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.

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]. Our AIWolf agents use the Japanese language, while the shared task organizers automatically translate the system I/O to connect with English agents.

1.1 The Werewolf Game

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 the team members to survive, not necessarily the player him/herself.

While there are many variations 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 werewolf team but if he/her is divined by a seer, then its result is human.

A game consists 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 werewolfs (a possessed may be alive), 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. RelatedWorks

There are many AIWolf agents that use machine learning. For example, [4] [5] estimate each player's role by SVM and neural network. However, it is difficult to add reasons of the estimation in such methods. As communication and persuasion is one of the key actions in the Werewolf game, reasons that can convince other players could control the game.

In addition, most of the machine learning agents estimate the role probability individually. However, it is more natural to estimate the entire set of roles, because information is limited in such an imperfect information game, estimation should be performed based on a chain of information.

For these reasons, we made a table that covers all the situations of inspection results, assuming that there are two players who come out as seers. Our agent utters logical inference results with reasons based on that table

3 Method

Figure 1 shows a flow of our proposed method.

3.1 Reasoning table of inspections and role combinations

A seer's behavior, both genuine and fake, in the first day is the most important source of information for determining each player's role; reasoning from their inspection results is important when a player needs a clear reason to persuade.

In this shared task's game setting (five players), a seer and a possessed often come out (CO) as a seer, and a werewolf pretends like a villager. It is empirically known that a werewolf pretends like a villager is advantageous for the werewolf; [6] reported that an agent implemented by reinforcement learning also behaved so. Thus, if two players come out as seers, we assume that they are a genuine seer and a possessed. Based on this assumption, we make a table that covers all possible variations of the inspected player's role. Since there are two villagers in this game setting, we also distinguished patterns, whether two seers inspected the same agent or not. We can cover all of the situations of seers' inspections by 20 patterns. From a corresponding situation pattern, we can assign reasons. Figure 2 and Table 1 show pattern examples. We made a subjective reason and an objective reason, corresponding to subjective (internal, hidden) and objective (external) point of view.

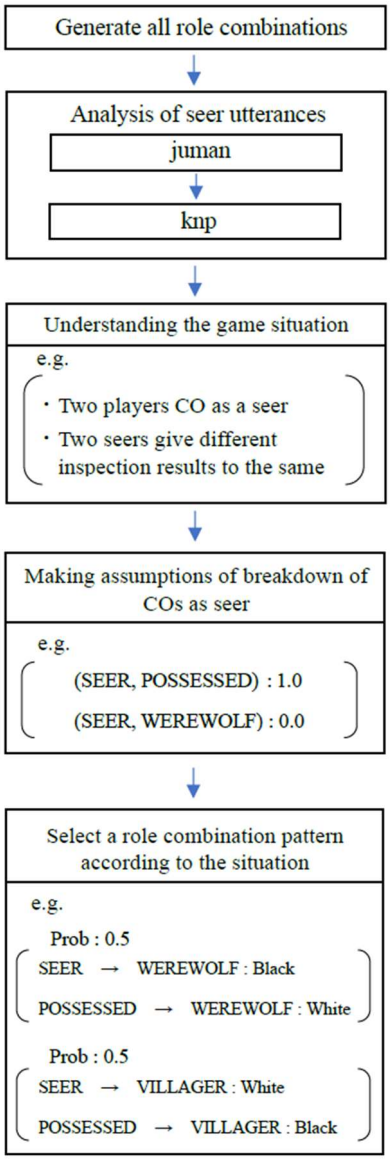

Figure 1 : The flow of our proposed method

[Pattern 17]

Agent01: SEER→POSSESSED(02) White
Agent02: POSSESSED→VILLAGER(04) Black
Agent03: WEREWOLF
Agent04: VILLAGER
Agent05: VILLAGER

Figure 2 : An example of role pattern

Viewpoint	Reasons
Objective	[Whichever is a seer, one can only be possessed from the inspection result], [werewolf is a player who does not CO as seer]
SEER	[I am a seer], [I inspect the other seer white]
POSSESSED(lie)	[I am a seer], [the agent I inspected black is werewolf]
WEREWOLF(lie)	[I am a villager], [werewolf is the player who has not CO other than me]
VILLAGER inspected	[Since I am a villager, the seer who inspected me is a possessed], [The wolf is a player who has not CO other than me]
VILLAGER uninspected	[I am a villager], [werewolf is the player who has not CO other than me]

Table 1 : Examples of reasons

3.2 Natural language analysis

We analyze a given natural language input to extract “come out as a seer”, “my inspection result is something”, etc. from utterances of other players. Then we try finding a corresponding pattern. This analysis is performed by converting input to our middle language expression [7] which based on [8]. Before this conversion, we pre-process the input by morphological analysis, dependency analysis, and case analysis. We use JUMAN [9] for the morphological analysis, KNP [10] for the dependency analysis and the case analysis. Figure 3 shows an example

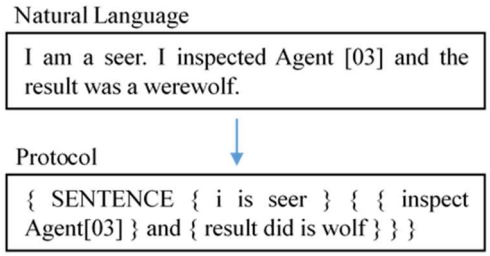

Figure 3 : An example of the middle language expression

3.3 Pattern selection

Even if we could analyze the utterance correctly, a given inspection situation may not correspond uniquely to one of the 20 patterns. For example, as shown in Figure 2, if a player, who has come out as a seer, gives an inspection result as white (human) to another player, and that another player gives an inspection result as black (werewolf) to the other player, then there are three possible cases:

1. (SEER → POSSESSED white), (POSSESSED → WEREWOLF black)
2. (SEER → POSSESSED white), (POSSESSED → VILLAGER black)
3. (SEER → WEREWOLF Black), (POSSESSED → SEER white)

The example above demonstrates an ambiguous case. This is because we can only distinguish situation patterns by whether the inspected player has come out as a seer or not. There are twelve possible situations in total, which should correspond to one of the 20 role combination patterns. Therefore, we have to assume disambiguation into one of the patterns, in addition to the assumption that "the breakdown of players, who made COs as seers, are a seer and a possessed". While we performed this disambiguation randomly in this paper, using machine learning and statistical information would be a future work.

3.4 Calculation of probability

Based on the assumptions described above, we can estimate probability of the role combinations for each player’s role as follows:

$$P\left(r_{i_j}\middle|c_k\right) = n_{i_j}/n_k$$

where r_{i_j} indicates that $agent_i$ is $role_j$. c_k stands for the game situation, n_k stands for the number of patterns that matches with c_k, n_{i_j} stands for the number of patterns that $agent_i$ is $role_j$ in n_k patterns. For example, when the pattern in Figure 2 is selected, the probability seen from Agent4 (VILLAGER) is as shown in Table 2.

Role	Agent1	Agent2	Agent3	Agent4	Agent5
SEER	1.0	0.0	0.0	0.0	0.0
POSSESSED	0.0	1.0	0.0	0.0	0.0
WEREWOLF	0.0	0.0	0.5	0.0	0.5
VILLAGER	0.0	0.0	0.5	1.0	0.5

Table 2 : Probabilities of each player’s role

4 Talk Example of Our Agent Implementation

We implemented our agent following the above method. Figure 4 shows a talk example of self-games where the same agents participate as all five players. The example shows that our agent performed reasoning and made utterances with consistent reasons. Simultaneously, our agent performs game actions such as vote and attack consistent with its utterances.

0,status,1,POSSESSED,ALIVE,Kanolab
0,status,2,VILLAGER,ALIVE,Kanolab
0,status,3,VILLAGER,ALIVE,Kanolab
0,status,4,WEREWOLF,ALIVE,Kanolab
0,status,5,SEER,ALIVE,Kanolab
…
1,talk,0,0,1,僕は占い師だね。Agent[02]を占ったら、結果は人狼だったね。
EN(I'm a seer. I inspected Agent[02] and the result is werewolf.)
1,talk,1,0,4,どうでもいいけどマカロン食べたい
EN(Somehow I want to eat macarons.)
1,talk,2,0,3,駆逐してやるぞ。
EN(I definitely kill a werewolf.)
1,talk,3,0,2,私は占い師ではないよ。
EN(I'm not a seer)
1,talk,4,0,5,あたいは占い師だな。Agent[03]を占ったけど、結果は人間だったぞ。
EN(I'm a seer. I inspected Agent[03] and the result is human.)
…
1,talk,5,1,2,私は村人で、自分に黒出しした占い師は狂人で、真占い師の占い結果から占われていない CO していない人が人狼で、私は Agent[04]に投票するね。
EN(I will vote agent04 because
I'm a villager and the seer who inspect me black is possessed and the player who has not CO is werewolf according to genuine seer's inspection.)
…
1,talk,25,5,3,俺は村人で、自分以外の CO していない人のどちらかが人狼だし、Agent[01]が狂人だと思うぞ。
EN(I think Agent[01] is a possessed because I'm a villager and the player who has not CO is werewolf.)

Figure 4 : Talk example of our Agent

5 Evaluation

AIWolfDial 2019 shared task organizers provided subjective evaluations. This subjective evaluation was performed according to the following criteria:(Table 3)

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 3 : The criteria for subjective evaluations

This subjective evaluation is based on both self-match games and mutual match games. The results are Table 4.

Name	Total	A	B	C	D	E
CanisLupus-JA	3.52	4	3.2	3.4	3.6	3.4
Dreaming-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

Table 4 : The evaluation results in AIWolfDial 2019

The evaluation items B, C, and D were relatively high for our agent. Regarding the evaluation item B, our agent could have inferred the roles reasonably from the inspections results. For example, in the fifth talk in figure 4, Agent[02] (villager) could correctly infer the roles of the other agents by assuming that a seer is fake, who inspected Agent[02] as a werewolf. Regarding the evaluation items C and D, our agent has kept consistency between utterances and game actions by using the role combination patterns. The advantage of our proposal method is as follows: once a game situation matches with a prepared pattern, we can keep high consistency by taking actions and generating utterances based on that pattern. On the other hand, our agent sometimes simply lists inference of roles, or repeats similar utterances may have made the lower evaluation results in A and E.

6 Conclusion and Future Work

We suggested a reasoning system for the Werewolf player using role patterns and heuristics. We implemented our agent based on this suggestion, participated the AIWolfDial 2019 shared task. Our agent could make inferences with clear reasons according to a given situation. There are two issues and potential future works as follows.

Firstly, our system relies on the results of natural language analysis. If the analysis is not performed correctly, the role estimation could fail. Such an incorrect analysis was often observed in the shared task.

Secondly, our reasoning table is not generic enough. We have to re-create the table when the game setting changes e.g. to a seven players' game. It is almost impossible to manually create the entire table when the number of players and roles get larger.

Determining the probabilities statistically from game logs would be a future work. Selecting patterns through communications with other agents is another option. To build a cooperative relationship between agents and take advantage of the games is the ultimate goal of our work, and we showed the first step for this goal in this paper.

Acknowledgments

We wish to thank the members of the Kano Laboratory in Shizuoka University who contributed to the valuable discussions. We thank Ms. Mukouyama, who made advices as an expert of the Werewolf game. 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] Kajiwara,K., Toriumi, F., Inaba, M., Osawa, H., katagami, D., Shinoda, K., Matsubara, H., Kano, Y., *"Development of AI Wolf Agent using SVM to Detect Werewolves"*, 2016, The 30th Annual Conference of the Japanese Society for Artificial Intelligence, (In Japanese)

[5] Okawa, T., Yoshinaka, R., Shinohara, A., *"Development of AI Wolf Agent Deducing Player's Role Using DeepLearning"*, 2017, The 22nd GameProgrammingWorkshop 2017, pp50-55, (In Japanese)

[6] Kajiwara, K., Toriumi, F., Osawa, H., Katagami, D., Inaba, M., Shinoda, K., Nishino, J., Ohashi, H., *"Abstraction of Optimal Strategy in "Are you a Werewolf?" by Reinforcement Learning"*, 2014, The 76th Information Processing Society of Japan, pp597-598, (In Japanese)

[7] Minowa, S., Takinami, A., Ogawa, C., Mihara, M., Maki, Y., Shiba, A., Kano, Y., *"Natural Language AIWolf Agent by Semantic Understanding Using Protocol"*, 2017, SIG-SLUD-81, The 8th Dialog system symposium, pp58-61, (In Japanese)

[8] Osawa, H., *"Communication Protocol for the "Werewolf"game"*, 2013, Human-Agent Interaction Symposium 2013, pp122-130, (In Japanese)

[9] Kurohashi, S., Kawahara, D., *Japanese morphological analysis system juman*, http://nlp.ist.i.kyoto-u.ac.jp/index.php?JUMAN

[10] Kurohashi, S., Kawahara, D., *Japanese syntax / case / anaphoric analysis system KNP*, http://nlp.ist.i.kyoto-u.ac.jp/index.php?KNP

[11] Shinoda, K., Toriumi, F., Katagami, D., Osawa, T., Inaba, M., *"Are you a Werewolf?" becomes a Standard Problem for General Artificial Intelligence"*, 2014, The 24th Intelligent system simposium, pp74-77, (In Japanese)

Strategies for an Autonomous Agent Playing the "Werewolf game" as a Stealth Werewolf

Shoji Nagayama[1] **Jotaro Abe**[1] **Kosuke Oya**[1] **Kotaro Sakamoto**[1]

Hideyuki Shibuki[2] **Tatsunori Mori**[1] **Noriko Kando**[2]

[1]Yokohama National University, Japan
[2]National Institute of Informatics, Japan
{nagayama, jotaro, kosuke-o, sakamoto, mori}@forest.eis.ynu.ac.jp
{shib, kando}@nii.ac.jp

Abstract

The "Werewolf game" is a popular multiplayer game wherein "villagers" try to figure out who is a "werewolf" through conversations. Werewolves usually pretend to be villagers. In this paper, we studied conversations in game logs in order to investigate how werewolves' cooperation contributed to increasing the winning percentage of the werewolves' team. As the number of "whispers" that are utterances via werewolves' private chat may be regarded as a measure of the werewolves' cooperation, we investigated the relation between the number of whispers and the winning percentage. As the result, we observed that the winning percentage of werewolves' team increased by 63 points at most when the number of whispers of at least two werewolves was more than 106.

1 Introduction

The "Werewolf game" is a popular multiplayer game wherein "villagers" try to figure out who is a "werewolf" through conversations. Werewolf game is actively researched, and competitions are also held in as shared task (Kano et al., 2019). As conversations in the game are open for all players, no player can talk with other players in secret. Therefore, working together with only his/her allies through conversations is difficult. Consequently, each player's thought and action become complicated. On the contrary, werewolves have a special talk channel, Whisper, through which they can secretly talk with other werewolves, allowing werewolves to work together. This is a strong advantage for werewolves, and it is an important factor so that werewolves win.

There are two basic strategies for werewolves. The first strategy is called as "swindle werewolf", wherein a werewolf makes himself/herself seem to be a leader of villagers, such as a seer or a medium. The second strategy is called as "stealth werewolf" wherein a werewolf hides himself/herself as one of the villagers. The swindle werewolf can have the initiative for misleading villagers, while it is easy to be a target of divination or execution. The stealth werewolf cannot have the initiative, but it is hard to raise a doubt of werewolf since he/she does not work directly on the subject of execution. We attempt to make the stealth werewolf an agent, and would like to clarify important factors for the stealth werewolf. If an agent can talk and mislead villagers without attracting attention from other players, it is a strong stealth werewolf. Although there are previous studies that have investigated conversations in the Werewolf game (Hirata et al., 2016), they are not done so from the standpoint of the stealth werewolf. Therefore, in this paper, we investigate the influences of the numbers of utterances, appearances in utterances of other players, and whispers, on the victory or defeat of werewolves.

2 Related work

There are the following previous researches about the Werewolf game. Toriumi et al., (2017) described the advantage of using the Werewolf game as "including the asymmetric diversity of player information, persuasion as a means of earning confidence, and speculation to detect fabrication." Gillespie et al.,(2016) used transcripts of the Werewolf game as the evaluation data of their semantic classifier. Takahashi et al.,(2017) measured trust between players through the arranged Werewolf game. Wang et al.,(2018) built a robot that had abilities such as casting a glance to play the real world Werewolf game. Xiong et al.,(2017) reported the optimal number of players to convey the attraction of the Werewolf game. The above researches did not aim to make agents in the Were-

Table 1: Number of data and players

players	files	the number of role						
		villager	seer	guard	medium	werewolf	possess	NPC
14	89	6	1	1	1	3	1	1
15	33	7	1	1	1	3	1	1
16	309	8	1	1	1	3	1	1
sum	431							

wolf game. Nide et al.,(2017) attempted to make a agent using extended BDI model, and conducted a thought experiment. However, no empirical experiment was conduct. Nakamura et al.,(2016) reported that estimating player roles based multiple perspectives increased winning rate. Hirata et al.,(2016) made an agent using action probabilities based on game logs of werewolf BBS for behaving like human beings. Their algorithms are not specialized in werewolf agent. We aim to construct a strategy for the stealth werewolf.

3 Werewolf BBS

Werewolf BBS is a bulletin board system for the online Werewolf game. A game session is called as "a village", which comprises 10 to 16 characters, including a non-player character (NPC)[1]. The game time synchronized with the real -world time, and it takes approximately a week to play a game. Each player can have up to 20 utterances per day. The non-verbal communication information is not allowed. As werewolves have a special talk channel, Whisper, they can discuss their strategy, for example, as to who takes charge of the swindle werewolf or the stealth werewolf.

In this study, we collected game logs from "Werewolf BBS: G villages" for analysis using Python library, Beautiful Soup[2]. We collected villages that included three werewolves, indicating that the number of villagers is 13 to 16, and of which players did not drop out, except execution or attack[3]. A village was collected as a file. Table 1 lists the number of collected files and the number of game roles in each village. There were no villages with 13 players that met the collection condition described above, and the total number of the files was 431 (243 MB). Although every file includes a prologue involving idle talk before the game roles are assigned to players, the prologue was excluded for analysis. The average number of utterances per file after excluding a prologue was 70.7.

4 Influence of utterances and appearances

Strong stealth werewolf talks into misleading villagers without attracting attention from other players. As judging whether an utterance can lead to misleading is difficult, we used the following two measures for attracting appearances.

The first measure is the number of utterances indicating how many times a player talks, because we considered that players with a lot of utterances were conspicuous. The second measure is the number of appearances indicating how many times a player comes up in utterances of other players, because we considered that it indicates how he/she attracts attention from other players. The more the number of utterances and appearances are, the more attention will be drawn.

Using decision trees, we analyzed how the numbers of utterances and appearances per player affected the winning percentage of werewolves. For making a decision tree of utterances, we used 16 character roles as attributes, the total numbers of utterances in a game as attribute values, and victory or defeat of werewolves as classes. The decision tree of appearance was made in the same manner. If a role such as werewolf or villager was assigned to two or more players, it was distinguished by the rank in the descending order of the number of utterances or appearances. If the number of players in a game was fewer than 16, we add dummy villagers to make up for the shortage. The utterance number of dummy villagers and the appearance number of those are assumed to be zero. We used the Python libraries scikit-learn[4] and dtreeviz [5] for making and showing decision trees, respectively. Figures 1 and 2 show the decision trees of utterances and appearances, respectively. Bifurcation occurs depending on a certain threshold for the number of utterances and ap-

[1]The number of actual players is 9 to 15.

[2]https://github.com/waylan/beautifulsoup

[3]Players who does not talk at least once a day are forcibly dropped out. Besides, players can stop playing the game of their own accord.

[4]https://github.com/scikit-learn/scikit-learn

[5]https: // github.com/parrt/dtreeviz

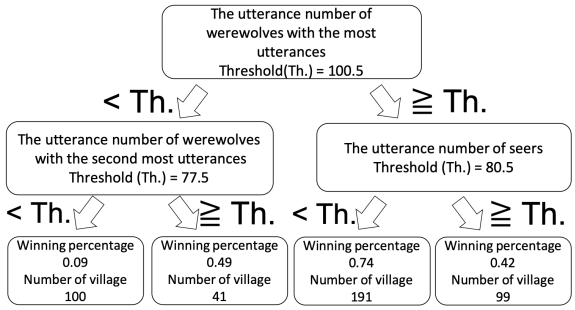

Figure 1: Decision tree based on the number of utterances

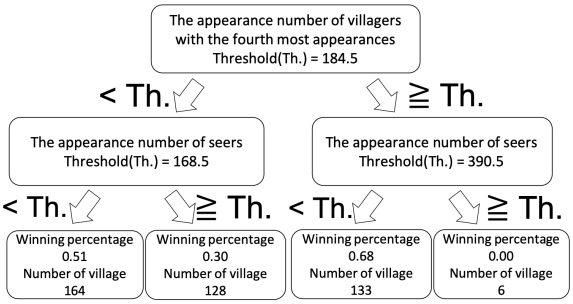

Figure 2: Decision tree based on the number of appearances

pearances. This corresponds to the case where the value of the right branch is greater than or equal to the threshold, and the case where the value of the left branch is less than the threshold. In the figures, "Winning percentage" represents the winning percentage of werewolves, and "Number of villages" represents the number of applicable villages. Each of leaves displays the winning percentage of werewolves and the number of applicable villages at the condition. From Figure 1, if the utterance number of the werewolf with the most utterances is 100.5 or more and the utterance number of seers is less than 80.5, the winning percentage of the werewolf teams is as high as 74 points across 191 villages.

Looking at Figure 1, it can be confirmed that the first branch is made by the utterance number of werewolves; thus, the victory or defeat branches depending on the utterance number of werewolves. There is also the utterance number of werewolves at the second branch, and if the utterance number of werewolves is less than a certain number, the winning percentage of the werewolves reduces. The winning percentage of werewolves at this time is at least 9 percent. If the utterance number of werewolves with the most utterance is more than the threshold and the number of utterance of the seer is fewer than the threshold, the winning percentage was increased from 9 to 74 points.

From Figure 2, the appearance number of villager with the fourth most appearances is the first branch, and the appearance number of seers is the second branch. It seems to be the subject of conversation, whether the particular villager is suspected of being a werewolf or whether the seer is real. Especially, it can be confirmed that if the appearance number of seer is low, the winning percentage of werewolves increases. However, the number of appearances of werewolves does not appear as a factor of victory or defeat, and it is difficult to reflect this knowledge on the specific tactics of werewolves.

Based on the above, we consider a method of manipulating the number of utterances of a specific player, centering on the utterance number of werewolves involved in victory or defeat. Especially when the werewolf manipulates the number of utterances of a specific player, it is necessary to cooperate well with other werewolves so that the operation does not suffer. The details of such a werewolf collaboration are explained in Section 5. The number of appearances is not as good as expected, and the effect of the number of appearances of werewolves on victory or defeat is not so great. In particular, when reducing the number of appearances, unlike the number of utterances, it cannot be controlled even if it is excluded from the game by execution or attack. For this reason, we will consider methods for estimating roles in which werewolves utter so as not to raise, the number of their appearance in utterance of other players.

5 Influence of whispers

One of the unique abilities of a werewolf that cannot be found in other roles is the "whisper" described in Section 3. Using "whisper" makes it possible for the werewolves to cooperate secretly, which can have a big influence on victory. For example, as described in Section 4, manipulating the utterance number of a specific player as a method called "asking" that affects victory or defeat is possible. "Asking" increases the utterance number of a specific player intentionally by seeking a response by speaking to a specific player. However,

as the number of utterances that a player can make per day is limited, controlling the number of utterances of all players alone is difficult. If our role is that of a werewolf, we may ask another werewolf who has sufficient room of utterances to use "asking" through whispers. The utterance number of specific players can be controlled such that the werewolf teams is advantageous. We investigated how the number of werewolves' whispers affects victory or defeat by making a decision tree. The decision tree for whispers was made in the same manner as that mentioned in Section 4. For making a decision tree of whispers, we used three werewolf players as attributes, the total numbers of whispers in a game as attribute values, and the victory or defeat of werewolves as classes.

In Figure 3, increasing the number of whispers does not simply means that the werewolves are cooperated well. For example, when a werewolf asks questions or proposes a strategy, another werewolf will not always get on his proposal. To get his proposal accepted, persuading through dialogue is necessary, which is the essence of the Werewolf game. In a dialogue, a response from another werewolf maybe expected for the utterance of a werewolf. If there is not much difference in the number of whispers of each werewolf, we may infer that the dialogue has been established. Therefore, we assume that the number of whispers among the werewolves is considerable, and the strategy and situation are well discussed and coordinated, if there is no difference in the whisper number of each werewolf.

From Figure 3, the first branch shows the winning percentage of werewolves is higher when the number of whispers is larger throughout the game. In the second right branch, the value of threshold is the number of whispers posted by the second most whispering werewolf. That means the winning percentage of werewolves is higher when two werewolves establish the dialogues frequently. Specifically, the winning percentage of werewolves is high at 67 percent when both the first branch and the second branch are above the threshold. The winning percentage increases by 63 points compared to the case where the first branch and the second branch are both below the threshold. However, this analysis does not evaluate the difference in the number of whispers from the viewpoint of the degree of cooperation between the werewolves, owing to which another

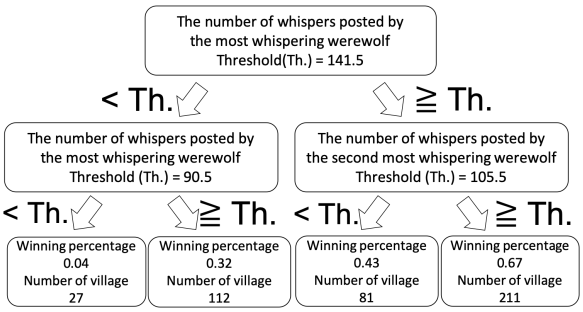

Figure 3: Decision tree based on the number of whispers

process is required. In addition, as the total of file size is only 40 MB, the number of villages used for analysis must be increased. However, when including villages where there are not more than three werewolves in the analysis, normalization is required because the number of roles and adopted roles set does not correspond to our current data set. In other words, investigating normalization conditions that do not depend on the number of werewolves and roles set is the immediate challenge.

6 Conclusion

In this paper, we analyzed 431 game logs of Werewolf BBS focusing on the "stealth werewolf", and confirmed that the winning percentage of werewolves increased by 65 points at most when the number of werewolf utterances was very frequent. We also confirmed that the winning percentage of werewolves increased by 63 points at most when the number of whispers was very frequent. In the future, we intend to proceed with research considering the content of utterances and whispers.

Acknowledgments

The research funded for / supported by the open collaborative research program at National Institute of Informatics (NII) Japan (FY2018).

References

Kellen Gillespie, Michael W. Floyd, Matthew Molineaux, Swaroop S. Vattam, and David W. Aha. 2016. Semantic Classification of Utterances in a Language-Driven Game. In *Communications in Computer and Information Science*, pages 116–129.

Yuya Hirata, Michimasa Inaba, Kenichi Takahashi, Fujio Toriumi, Hirotaka Osawa, Daisuke Katagami, and Kosuke Shinoda. 2016. Werewolf Game Modeling using Action Probabilities based on Play Log Analysis. In *9th International Conference on Computers and Games*, pages 103–114.

Yoshinobu Kano, Claus Aranha, Michimasa Inaba, Hirotaka Osawa, Daisuke Katagami, Takashi Otsuki, and Fujio Toriumi. 2019. Overview of the AIWolf-Dial 2019 Shared Task: Competition to Automatically Play the Conversation Game "Mafia". In *In proceedings of the 1st International Workshop of AI Werewolf and Dialog System (AIWolfDial 2019), the 12th International Conference on Natural Language Generation (INLG 2019)*.

Noritsugu Nakamura, Michimasa Inaba, Kenichi Takahashi, Fujio Toriumi, Hirotaka Osawa, Daisuke Katagami, and Kosuke Shinoda. 2016. Constructing a Human-like agent for the Werewolf Game using a psychological model based multiple perspectives. In *2016 IEEE Symposium Series on Computational Intelligence*. IEEE.

Naoyuki Nide and Shiro Takata. 2017. Tracing Werewolf Game by Using Extended BDI Model. In *Special Section on Frontiers in Agent-based Technology*, pages 2888–2896.

Hideyuki Takahashi, Midori Ban, Hirotaka Osawa, Junya Nakanishi, Hidenobu Sumioka, and Hiroshi Ishiguro. 2017. Huggable Communication Medium Maintains Level of Trust during Conversation Game. In *Frontiers in psychology*.

Fujio Toriumi, Hirotaka Osawa, Michimasa Inaba, Daisuke Katagami, Kosuke Shinoda, and Hitoshi Matsubara. 2017. AI Wolf Contest -Development of Game AI Using Collective Intelligence-. In *Communications in Computer and Information Science*, pages 101–115.

Bohao Wang, Hirotaka Osawa, Takuya Toyono, Fujio Toriumi, and Daisuke Katagami. 2018. Development of Real-World Agent System For Werewolf Game. In *17th International Conference on Autonomous Agents and Multiagent Systems*, pages 1838–1840.

Shuo Xiong, Wenlin Li, Xinting Mao, and Hiroyuki Iida. 2017. Mafia Game Setting Research using Game Refinement Measurement. In *14th International Conference Advances in Computer Entertainment Technology*, pages 830–846.