

Evaluation of a Fully Automatic Cooperative Persuasive Dialogue System

Takuya Hiraoka[†], Graham Neubig[†], Sakriani Sakti[†], Tomoki Toda[†], Satoshi Nakamura[†]

Abstract In this paper, we construct and evaluate a fully automated text-based cooperative persuasive dialogue system, which is able to persuade the user to take a specific action while maintaining user satisfaction. In our previous works, we created a dialogue management module for cooperative persuasive dialogue [1], but only evaluated it in a wizard-of-Oz setting, as we did not have the capacity for natural language generation and understanding. In this work, the main technical contribution is the design of the natural language understanding (NLU) and the natural language generation (NLG) modules which allows us to remove this bottleneck and create the first fully automatic cooperative persuasive dialogue system. Based on this system, we performed an evaluation with real users. Experimental results indicate that the learned policy is able to effectively persuade the users: the reward of the proposed model is much higher than baselines, and almost the same as a dialogue manager controlled by a human. This tendency is almost the same as our previous evaluation using a wizard-of-Oz framework [1] demonstrating that the proposed NLU and NLG modules are effective for cooperative persuasive dialogue.

1 Introduction

There is ongoing research on applying reinforcement learning to persuasion and negotiation dialogues, which are different from traditional task-based dialogues [2, 3, 4, 5]. In task-based dialogue, the system is required to perform the dialogue to achieve the user goal, eliciting some information from the user to provide an appropriate service. A reward corresponding to the achievement of the user’s goal is given to the system. In contrast, in persuasive dialogue, the system convinces the user to take some action achieving a system goal, for example buying a particular product or agreeing to a particular plan [3]. In previous work, we have proposed the paradigm of cooperative persuasive dialogue [1, 6], where reward corresponding to the achievement of both the user’s and the system’s goal is given to the system. This paradigm is useful in situations where the user and system have different, but not mutually exclusive goals, an example of which being a sales situation where the user wants to find a product that matches their taste, and the system wants to successfully sell a product, ideally one with a higher profit margin.

In previous reports, we have applied reinforcement learning to cooperative persuasive dialogue, and evaluated the learned policy in a wizard-of-Oz setting [1]. We modeled the cooperative dialogue based on partially observable Markov decision

[†]Nara Institute of Science and Technology (Japan)
e-mail: {takuya-h,neubig,ssakti,tomoki,s-nakamura}@is.naist.jp

processes (POMDP), and system policies were learned with reinforcement learning. We introduced framing [7], description of alternatives with emotionally charged words, as a system action. In this previous work, we evaluated the learnt policy by substituting a human wizard of Oz for natural language understanding (NLU) and the natural language generation modules (NLG). In this evaluation framework, the result of the evaluation is highly dependent on the ability of the human wizard, and the effect of NLU and NLG is discounted, potentially overstating the effectiveness of the system.

In this paper, we construct and evaluate the first fully automated text-based cooperative persuasive dialogue system. At first, we give a review of our previous research [1, 8] about learning cooperative persuasive policies, and then explain new modifications to the dialogue modeling, the newly implemented NLU and NLG models, and the evaluation. Experimental results indicate that the learned policy with framing is effective, even in a fully automatic system. The reward of the learnt policy with framing is much higher than baselines (a policy without framing, and a random policy), and almost the same as a policy controlled by a human. This tendency is almost the same as the result of our previous research using the wizard-of-Oz framework [1].

2 Cooperative persuasive dialogue corpus

In this section, we give a brief overview of cooperative persuasive dialogue, and a human dialogue corpus that we use to construct the dialogue models and dialogue system described in later sections. In our collected persuasive dialogue corpus (Section 2.1), we define and quantify the actions of the cooperative persuader (Section 2.2). In addition, we annotate persuasive dialogue acts of the persuader from the point of view of framing (Section 2.3).

2.1 *Persuasive dialogue corpus*

The cooperative persuasive dialogue corpus [8] consists of dialogues between a salesperson (persuader) and customer (persuadee) as a typical example of persuasive dialogue. The salesperson attempts to convince the customer to purchase a particular product (decision) from a number of alternatives (decision candidates). More concretely, the corpus assumes a situation where the customer is in an appliance store looking for a camera, and the customer must decide which camera to purchase from 5 alternatives.

Prior to recording, the salesperson is given the description of the 5 cameras and instructed to try to convince the customer to purchase a specific camera (the persuasive target). In this corpus, the persuasive target is camera A, and this persuasive target is invariant over all subjects. The customer is also instructed to select one preferred camera from the catalog of the cameras¹, and choose one aspect of the camera that is particularly important in making their decision (the determinant). During recording, the customer and the salesperson converse and refer to the infor-

¹ The salesperson is not told this information about customer preferences.

mation in the camera catalog as support for their dialogues. The customer can close the dialogue whenever they want, and choose to buy a camera, not buy a camera, or reserve their decision for a later date. The total number of dialogues is 34, and the total time is about 340 minutes.

2.2 Annotation of persuader and persuadee goals

We define the cooperative persuader as a persuader who achieves both the persuader and persuadee goals, and cooperative persuasive dialogue as a dialogue where both the persuader and persuadee goals have been achieved. To measure the salesperson's success as a cooperative persuader, we annotate each dialogue with scores corresponding to the achievement of the two participants' goals. As the persuader's goal, we use persuasive success measured by whether the persuadee's final decision (purchased camera) is the persuasive target or not. As the persuadees goal, we use the persuadee's subjective satisfaction as measured by results of a questionnaire filled out by the persuadee at the end of the dialogue (1: Not satisfied 3: Neutral 5: Satisfied). Note that we assume a situation that is not a zero-sum game, and thus the persuader and persuadee goals are not mutually exclusive.

2.3 Annotation of dialogue acts

2.3.1 Framing

Framing is the use of emotionally charged words to explain particular alternatives, and is known as an effective way of increasing persuasive power. The corpus contains tags of all instances of negative/positive framing [7, 9], with negative framing using negative words and positive framing using positive words.

The framing tags are defined as a tuple $\langle a, p, r \rangle$ where a represents the target alternative, p takes value NEG if the framing is negative, and POS if the framing is positive, and r is a binary variable indicating whether or not the framing contains a reference to the determinant that the persuadee indicated was most important (for example, the performance or price of a camera). The user's preferred determinant is annotated based on the results of the pre-dialogue questionnaire.

Table 1 shows an example of positive framing ($p=POS$) about the performance of Camera A ($a=A$). In this example, the customer answered that his preference is the price of camera, and this utterance does not contain any description of price. Thus, $r=NO$ is annotated.

Table 1 An example of positive framing

(Camera A is) able to achieve performance of comparable single-lens cameras and can fit in your pocket, this is a point.
--

2.3.2 General purpose functions (GPF)

The corpus also contains tags for traditional dialogue acts. As a tag set to represent traditional dialogue acts, we use the general-purpose functions (GPF) defined by the ISO international standard for dialogue act annotation [10]. All annotated GPF tags are defined to be one of the tags in this set.

3 Cooperative persuasive dialogue modeling

The cooperative persuasive dialogue model proposed in our previous research [1] consists of a user-side dialogue model (3.1) and a system-side model (3.2).

3.1 User simulator

The user simulator estimates two aspects of the conversation:

1. The user’s dialogue acts.
2. Whether the preferred determinant has been conveyed to the user (conveyed preferred determinant; CPD).

The users’ dialogue acts are represented by using GPFs (e.g. question, answer, and inform). In our research, the user simulator chooses one GPF or *None* representing no response at each turn. CPD represents that the user has been convinced that the determinant in the persuader’s framing satisfies the user’s preference. For example, in Table 1, “performance” is contained in the salesperson’s positive framing for camera A. If the persuadee is convinced that the decision candidate satisfies his/her preference based on this framing, we say that CPD has occurred ($r=$ YES). In our research, the user simulator models CPD for each of the 5 cameras. This information is required to calculate reward described in the following Section 3.2. Specifically, GPF and CPD are used for calculating naturalness and persuasive success, which are elements of the reward function.

The user’s GPF G_{user}^{t+1} and CPD C_{alt}^{t+1} at turn $t + 1$ are calculated by the following probabilities:

$$P(G_{user}^{t+1} | G_{user}^t, F_{sys}^t, G_{sys}^t, U_{eval}) \quad (1)$$

$$P(C_{alt}^{t+1} | C_{alt}^t, F_{sys}^t, G_{sys}^t, U_{eval}). \quad (2)$$

G_{sys}^t represents the system GPF at time t , and F_{sys}^t represents the system framing at t . These variables correspond to system actions, and are explained in Section 3.2. G_{user}^t represents the user’s GPF at t , C_{alt}^t represents the CPD at t , and U_{eval} represents the users’s original evaluation of the alternatives². In our research, this is the camera selected by the user as preferred at the beginning of the dialogue. We use the persuasive dialogue corpus described in Section 2.1 for training the user simulator, considering the customer in the corpus as the user and the salesperson in the corpus as the system. We use logistic regression for learning Equations (1) and (2).

3.2 Dialogue modeling: learning cooperative persuasion policies

For training the dialogue system using reinforcement learning, in addition to the user simulator, the reward, system actions, and belief state are required [11].

Reward is calculated using three factors: user satisfaction, system persuasive success, and naturalness. As described in Section 1, cooperative persuasive dialogue

² Values of these variables are set at the beginning of dialogue, and invariant over the dialogue.

Table 2 Features for calculating reward. These features are also used as the system belief state.

Sat_{user}	Frequency of system commissives	PS_{sys}	Total time	N	System and user current GPF
	Frequency of system question		C_{alt} (for of 6 cameras)		System and user previous GPF
			U_{eval} (for of 6 cameras)		System framing

systems must perform dialogue to achieve both the system and user goals. Thus, reward at each turn t is calculated with the following equation:

$$r_t = (Sat_{user}^t + PS_{sys}^t + N^t)/3. \quad (3)$$

Sat_{user}^t represents a 5 level score of the user’s subjective satisfaction (1: Not satisfied, 3: Neutral, 5: Satisfied) at turn t scaled into the range between 0 and 1. PS_{sys}^t represents persuasive success (1: SUCCESS, 0: FAILURE) at turn t . N_t represents bi-gram likelihood of the dialogue between the system and user at turn t . Sat and PS are calculated with a predictive model constructed from the corpus described in Section 2.1 [8].

The **system action** $\langle G_{sys}, F_{sys} \rangle$ is a GPF/framing $\langle a, p \rangle$ pair representing the dialogue act of the salesperson. We construct a unigram model of the salesperson’s dialogue acts $P(G_{sales}, F_{sales})$ from the original corpus, then exclude pairs for which the likelihood is below 0.005. As a result, we use the remaining 13 pairs as system actions.

The **belief state** is represented by the features used for reward calculation (Table 2) and the reward calculated at previous turn. Note that of the 8 features used for reward calculation, only C_{alt} can not be directly observed from the system action or NLU results, and thus the system estimates it through the dialogue by using the following probability:

$$\sum_{C_{alt}^t} P(C_{alt}^{t+1} | C_{alt}^t, F_{sys}^t, G_{sys}^t, U_{eval}) P(C_{alt}^t), \quad (4)$$

where C_{alt}^{t+1} represents the estimated CPD at $t + 1$, C_{alt}^t represents the estimated CPD at t , and the other variables are the same as those in Equation (2).

4 Modifications of the cooperative persuasive dialogue model

In this paper, we further propose two modifications to the cooperative dialogue models described in Section 3: 1) considering NLU recognition errors in the belief state, and 2) normalization of reward factors.

4.1 Considering NLU recognition errors

In the cooperative dialogue model in Section 3, we are not considering recognition errors of the NLU module. In previous research [1], we evaluated the policies based on the wizard of Oz, where a human was substituted for the NLU module, precluding the use of estimation methods used in ordinary POMDP-based dialogue systems [11]. However, in this paper, we use a fully automatic NLU module, which might cause recognition errors, and thus some method for recovery is needed.

In this work, we modify the dialogue model to consider NLU recognition errors, incorporating estimation of the true user dialogue act (i.e. GPF) into the dialogue model. The estimation is performed according to the following equation:

$$P(G_{user}^{t+1}|H_{G_{user}}) = \frac{\sum_{G_{user}^t} P(H_{G_{user}^{t+1}}|G_{user}^{t+1})P(G_{user}^{t+1}|G_{user}^t)P(G_{user}^t)}{\sum_{G_{user}^{t+1}} \sum_{G_{user}^t} P(H_{G_{user}^{t+1}}|G_{user}^{t+1})P(G_{user}^{t+1}|G_{user}^t)P(G_{user}^t)}. \quad (5)$$

H_{user} represents the NLU result (described in Section 5.1) at t , and other variables are the same as those in Eqs. (1) and (2). $P(H_{G_{user}^{t+1}}|G_{user}^{t+1})$ represents a confusion matrix between the actual GPF and recognition result. To construct the confusion matrix, in Section 6.1, we perform an evaluation of NLU and use the confusion matrix from this evaluation for the estimation of Eq. (5). $P(G_{user}^{t+1}|G_{user}^t)$ is calculated using maximum likelihood estimation over the persuasive dialogue corpus described in Section 2.1.

4.2 Normalization of the reward factors

The reward function in Section 3.2 considers three factors: persuasive success, user satisfaction, and naturalness. In the current phase of our research, we have no evidence that one of these factors is more important than the other for cooperative persuasive dialogue, and thus would like to treat them as equally important. However, in Eq. (3) the scales (i.e. the standard deviation) of factors are different, and thus factors with a larger scale are considered as relatively important, and other factors are considered as relatively unimportant. For example, in our previous research [1], the scale of naturalness N is smaller than other factors, and as a result is largely ignored in the learning.

In this work, we fix this problem by equalizing the importance of reward factors through normalization with z-score. More concretely, the reward function of Eq (3) is substituted with the following reward function:

$$r'_t = \frac{Sat_{user}^t - \overline{Sat_{user}^t}}{\text{Stddev}(Sat_{user})} + \frac{PS_{sys}^t - \overline{PS_{sys}}}{\text{Stddev}(PS_{sys})} + \frac{N^t - \overline{N}}{\text{Stddev}(N)}, \quad (6)$$

where variables with a bar represent the mean of variables without a bar, and the Stddev function represents the standard deviation of the argument. These statistics are calculated from simulated dialogue with the proposed dialogue model in the previous section, where actions are chosen randomly. We sampled the reward factor for 60,000 turns of the simulated dialogue (about 6000 dialogues) for calculating the statistics of each variable.

5 Text-based cooperative persuasive dialogue system

The main contribution of this paper is the construction of a fully automated text-based cooperative persuasive dialogue system. The structure of the system is shown in Figure 1. In this section, we describe the construction of NLU (Section 5.1) and

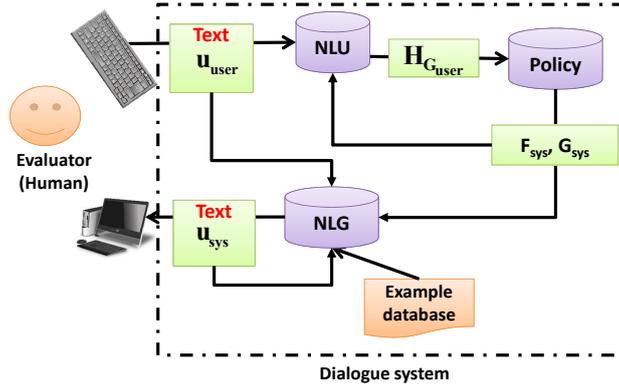


Fig. 1 Structure of our dialogue system. Rectangles represent information, and cylinders represent a system module.

NLG (Section 5.2) modules that act as an interface between the policy module and the human user, and are necessary for fully automatic dialogue.

5.1 Natural language understanding

The NLU module detects the GPF in the user's text input u_{user} using a statistical classifier. In this paper, we use bagging, using decision trees as the weak classifier [12]. We require the NLU to 1) be simple and 2) output the estimated classes with probability, and bagging with decision trees satisfies these requirements. The NLU uses many features (i.e. word frequency), and decision trees can select a small number of effective features, making a simple classifier. In addition, by using bagging, the confidence probability, which is determined by the voting rate of decision trees, can be attached to the classification result. We utilize Weka [13] for constructing the bagging classifier.

As input to the classifier, we use features calculated from u_{user} and the history of system outputs $(u_{sys}, \langle G_{sys}, F_{sys} \rangle)$. Features are mainly categorized into 4 types:

- Uni:** Unigram word frequency in the user's input.
- Bi:** Bigram word frequency in the user's input.
- DAcl:** The previous action of the system (i.e. GPF/framing pairs $\langle G_{sys}, F_{sys} \rangle$).
- Unicl:** Unigram word frequency in the previous system utterance.

As we use Japanese as our target language, we perform morphological analysis using Mecab [14], and use information about the normal form of the word and part of speech to identify the word.

As the NLU result $H_{G_{user}}$, 8 types of GPF are output with membership probabilities. We use 694 customer utterances in the camera sales corpus (Section 2) as training data. In this training data, 8 types of GPF labels are distributed as shown in Table 3.

Table 3 Distribution of the GPF labels in the training data.

Other	Question	SetQuestion	PropositionalQuestion	Inform	Answer	Directive	Commissive
46	4	12	156	260	117	36	63

5.2 Natural language generation

The NLG module outputs a system response u_{sys} based on the user’s input u_{user} , the system’s previous utterance u'_{sys} and the system action $\langle G_{sys}, F_{sys} \rangle$. Though the dialogue assumed in this paper is focusing on a restricted situation, it is still not trivial to create system responses for various inputs. In order to avoid the large amount of engineering required for template-based NLG and allow for rapid prototyping, we decide to use the framework of example-based dialogue management [15].

We construct an example database $D = \{d_1, d_2, \dots, d_M\}$ with M utterances by modifying the human persuasive dialogue corpus of Section 2. In the example database, the i th datum $d_i = \langle s, u, g, f, p \rangle$ consists of the speaker s , utterance u , GPF g , framing flag f , and previous datum p . In modifying the human persuasive dialogue corpus, we manually make the following corrections:

- Deletion of redundant words and sentences (e.g. fillers and restatements).
- Insertion of omitted words (e.g. subjects or objects) and sentences.

Our example database consists of 2022 utterances (695 system utterances and 1327 user example utterances). An example of the database is shown in Table 4.

Table 4 Part of the example database. The words surrounded by $\langle \rangle$ are inserted in correction.

Speaker	Utterance	GPF	Framing
User	I want camera A. Do you have it? (私はAのカメラが欲しいんですけどありますか?)	PropQ	
Sys.	Yes, we do have \langle A camera in the store \rangle . (\langle Aのカメラは店に \rangle ありますよ)	Answer	
Sys.	What was the good point of camera A? (Aのカメラのどこがよかったんですか?)	Question	
User	Well, I like its shape, like a Monolith. (そうですね。このモノリスみたいな露骨な形が好だからです)	Answer	
Sys.	The main difference between camera A \langle and other cameras \rangle is the sensor. (Aのカメラ \langle と他のカメラの大きな \rangle 違いはセンサーです) It is said that sensors are essential for a digital camera. (デジタルカメラはセンサーが命といわれています) The sensor of camera A is the same as that as a single-lens cameras. (Aのカメラのセンサーは一眼と同じセンサーを使ってるんですね。)	Inform	Pos A

The NLG module determines the system response u_{sys} based on u_{user} , u'_{sys} , and $\langle G_{sys}, F_{sys} \rangle$. More concretely, our NLG modules performs the following procedure:

1. We define the response candidate set R according to whether there is user input ($u_{user} \neq \phi$) or not ($u_{user} = \phi$). If $u_{user} \neq \phi$, then we define R as the set of utterances r for which the previous utterance is a user utterance ($r.p.s = User$). Conversely, if $u_{user} = \phi$, then we define R so $r.p.s = Sys$ ³.
2. Response candidates R are scored based on the following similarity score

³ In this paper, we use “.” for representing the membership relation between variables. For example, $Var1.Var2$ means that $Var2$ is a member variable of $Var1$.

$$\cos(r.p.u, u_{input}) = \frac{\text{words}(r.p.u) \cdot \text{words}(u_{input})}{|\text{words}(r.p.u)| \cdot |\text{words}(u_{input})|} \quad (7)$$

$$u_{input} = \begin{cases} u'_{sys} & (u_{user} = \phi) \\ u_{user} & (u_{user} \neq \phi). \end{cases}$$

The cosine similarity \cos between the previous utterance of the response sentence candidate $r.p.u (r \in R)$ and input sentence u_{input} is used for the scoring. u_{input} is set as u'_{sys} or u_{user} depending on u_{user} . The words function returns the frequency vector of the content words (i.e. nouns, verbs, and adjectives) weighted according to tf-idf.

3. The $r^*.u$ that has the highest score is selected as the output of the NLG module u_{sys}

$$r^* = \arg \max_{r \in R} \cos(r.p.u, u_{input}) \quad (8)$$

$$u_{sys} = r^*.u. \quad (9)$$

6 Experimental results

In this section, we perform two forms of experimental evaluation. First, as a preliminary experiment, we evaluate the performance of the NLU module proposed in Section 5.1. Then, we evaluate the fully automatic persuasive dialogue system.

6.1 Evaluation for NLU using different feature sets

First, we evaluate the performance of the NLU module using different feature sets proposed in Section 5.1. We prepare 4 patterns of feature sets (Uni, Uni+DAcl, Uni+CAcl+Unicl and Uni+CAcl+Bi), and evaluate the recognition accuracy of GPF labels in the customer's utterances. The evaluation is performed based on 15-fold cross-validation with 694 customer utterances described in Section 5.1.

From the experimental result (Figure 2), we can see that NLU with Uni+CAcl+Bi achieves the highest accuracy, and thus we decided to use Uni+CAcl+Bi for NLU of the dialogue system in the next section. Focusing on the details of the misclassified GPFs, we show the confusion matrix for classification results of the NLU module with Uni+CAcl+Bi in Table 5. From this matrix, we can see that Answer is misclassified to Inform, and that SetQ and Question are misclassified into PropositionalQ. This result indicates that this module has difficulty in distinguishing dialogue acts in a hypernym/hyponym or sibling relationship.

6.2 Complete system evaluation

In this section, we describe the results of the first user study evaluating fully automated cooperative persuasive dialogue systems.

For evaluation, we prepare the following 4 policies.

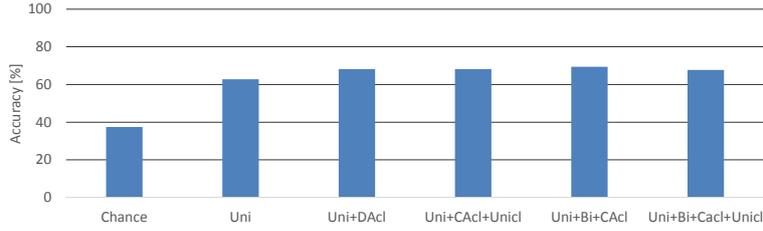


Fig. 2 Accuracy of the NLU module. The vertical axis represents accuracy and the horizontal axis represents the NLU feature set. Chance rate is an NLU module that always outputs Inform.

Table 5 The confusion matrix. Each row represents the distribution of the true GPF label. Each column represents the distribution of the NLU classification result.

Other	Commissive	PropQ	Directive	Answer	Inform	SetQ	Question	Classified as/True label
43	0	0	0	0	3	0	0	Other
6	31	2	4	0	20	0	0	Commissive
0	1	112	3	0	40	0	0	PropQ
2	2	6	13	0	13	0	0	Directive
0	3	5	0	53	56	0	0	Answer
1	12	4	4	9	230	0	0	Inform
0	0	10	0	0	2	0	0	SetQ
0	0	3	0	0	1	0	0	Question

Random: A baseline where the action is randomly output from all possible actions.

NoFraming: A baseline where the action is output based on the policy which is learned using only GPFs. For constructing the actions, we remove actions whose framing is not *None* from the actions described in Section 3.2. The policy is a greedy policy, and selects the action with the highest score.

Framing: The proposed method where the action is output based on the policy learned with all actions described in Section 3.2 including framing. The policy is also a greedy policy.

Human: An oracle where the action is output based on human selection. In this research, the first author (who has no formal sales experience, but experience of about 1 year in analysis of camera sales dialogue) selects the action.

For learning the policies (i.e. NoFraming and Framing), we use Neural fitted Q Iteration (NFQ) [16]. For applying NFQ, we use the Pybrain library [17]. The learning conditions follow the default Pybrain settings. We consider 3,000 dialogues as one epoch, and update the parameters of the neural network at each epoch. Learning is finished when number of epochs reaches 20 (60,000 dialogues), and the policy with the highest average reward is used for evaluation.

We evaluate policies on the basis of average reward and correct response rate of dialogues with real users. The definition of the reward is described in Section 3.2, and the correct response rate is the ratio of correct system responses to all system responses. In the experiment, the dialogue system plays the salesperson, and the user plays the customer. At the end of the dialogue, to calculate the reward, the user answers the following questionnaire:

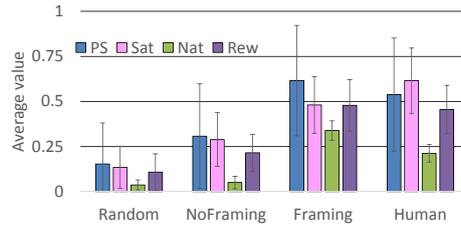


Fig. 3 Evaluation results for real users. Error bars represent 95% confidence intervals. Rew represents the reward, Sat represents the user satisfaction, PS represents persuasive success, and Nat represents naturalness.

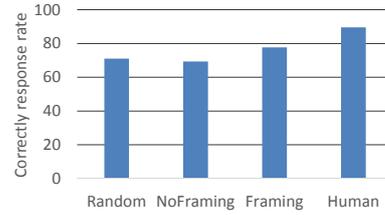


Fig. 4 Correct response rate of the system utterances.

Satisfaction: The user’s subjective satisfaction defined as a 5 level score of customer satisfaction (1: Not satisfied, 3: Neutral, 5: Satisfied).

Final decision: The camera that the user finally wants to buy.

In addition, to calculate the correct response rate, we have the user annotate information regarding whether each system response is correct or not. 13 users perform one dialogue with the system obeying each policy (a total of 4 dialogues per users).

Experimental results for the reward are shown in Figure 3. From these results, we can see that the reward of Framing is higher than that of NoFraming and Random, and almost equal to Human. This indicates that learning a policy with framing is effective in a fully automatic text-based cooperative dialogue system. It is interesting to note that the tendency of those scores is almost the same as those of the wizard-of-Oz based experiment [1]. The exception is that the naturalness of Framing in this experiment is higher than that of the wizard-of-Oz based experiment. Our hypothesis about the reason for this difference is that this is due to the effect of the modification of reward factors. In Section 4.2, we modified the importances of reward factors to be considered equally in learning the policy. Therefore, in the learning, naturalness is considered as an important factor, resulting in an increase of the naturalness score of Framing. It should be noted, however, that most of the subjects are different from the wizard-of-Oz based experiment we performed in previous work [1], and this might also affect the experimental result.

Experimental results for the correct response rate (Figure 4) indicate that our cooperative persuasive dialogue system somewhat correctly responds to the user’s input. The scores of all policies are higher than 70%, and the score of Framing is about 77%. In addition, even the Random policy achieves a score of about 70%. One of the reasons for this is that NLG method used by our system (Section 5.2) is based on examples, and thus is able to return natural responses that will only be judged as incorrect if they do not match the context.

7 Conclusion

In this paper, we presented a method for construction of a fully automatic cooperative persuasive dialogue system. Particularly, we focused on modifications to the policy learning, and construction of NLU and NLG modules. We performed an evaluation of the constructed dialogue system with real users. Experimental results

indicated that the proposed system is effective in text-based cooperative dialogue systems, and that the tendency of each reward is almost the same as results of our previous research [1].

In the future, we plan to evaluate the system policies in more realistic situations, that move beyond role-playing to real sales situations over more broad domains. We also plan to consider non-verbal information for estimating persuasive success and user satisfaction.

References

1. Takuya Hiraoka, Graham Neubig, Sakriani Sakti, Tomoki Toda, and Satoshi Nakamura, “Reinforcement learning of cooperative persuasive dialogue policies using framing,” *Proceedings COLING*, 2014.
2. Kallirroi Georgila and David Traum, “Reinforcement learning of argumentation dialogue policies in negotiation,” *Proceedings of INTERSPEECH*, 2011.
3. Kallirroi Georgila, “Reinforcement learning of two-issue negotiation dialogue policies,” *Proceedings of the SIGDIAL*, 2013.
4. Praveen Paruchuri, Nilanjan Chakraborty, Roie Zivan, Katia Sycara, Miroslav Dudik, and Geoff Gordon, “POMDP based negotiation modeling,” *Proceedings of the first MICON*, 2009.
5. Peter A. Heeman, “Representing the reinforcement learning state in a negotiation dialogue,” *Proceedings of ASRU*, 2009.
6. Takuya Hiraoka, Yuki Yamauchi, Graham Neubig, Sakriani Sakti, Tomoki Toda, and Satoshi Nakamura, “Dialogue management for leading the conversation in persuasive dialogue systems,” *Proceedings of ASRU*, 2013.
7. Levin Irwin, Sandra L. Schneider, and Gary J. Gaeth, “All frames are not created equal: A typology and critical analysis of framing effects,” *Organizational Behavior and Human Decision Processes* 76.2, 2013.
8. Takuya Hiraoka, Graham Neubig, Sakriani Sakti, Tomoki Toda, and Satoshi Nakamura, “Construction and analysis of a persuasive dialogue corpus,” *Proceedings of IWSDS*, 2014.
9. Irene Mazzotta and Fiorella de Rosis, “Artifices for persuading to improve eating habits,” *AAAI Spring Symposium: Argumentation for Consumers of Healthcare*, 2006.
10. ISO24617-2, *Language resource management-Semantic annotation frame work (SemAF), Part2: Dialogue acts*. ISO, 2010.
11. Jason D. Williams and Steve Young, “Partially observable Markov decision processes for spoken dialog systems,” *Computer Speech and Language*, 2007.
12. Leo Breiman, “Bagging predictors,” *Machine Learning*, 1996.
13. Weka 3: Data Mining Software in Java, “<http://www.cs.waikato.ac.nz/ml/weka/>” .
14. Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto, “Applying conditional random fields to Japanese morphological analysis.,” *Proceedings of EMNLP*, 2004.
15. Cheongjae Lee, Sangkeun Jung, Seokhwan Kim, and Gary Geunbae Lee, “Example-based dialog modeling for practical multi-domain dialog system.,” *Speech Communication*, 2009.
16. Martin Riedmiller, “Neural fitted Q iteration - first experiences with a data efficient neural reinforcement learning method,” *Machine Learning: ECML*, 2005.
17. Tom Schaul, Justin Bayer, Daan Wierstra, Yi Sun, Martin Felder, Frank Sehnke, Thomas Rückstieß, and Jürgen Schmidhuber, “Pybrain,” *The Journal of Machine Learning Research*, 2010.