Constructing a Non-task-oriented Dialogue Agent using Statistical Response Method and Gamification

Michimasa Inaba[1], Naoyuki Iwata[2], Fujio Toriumi[3], Takatsugu Hirayama[2], Yu Enokibori[2], Kenichi Takahashi[1] and Kenji Mase[2]

[1]*Graduate School of Information Sciences, Hiroshima City University, 3-4-1 Ozukahigashi, Asaminami-ku, Hiroshima, Japan*

[2]*Graduate School of Information Sciences, Nagoya University, Furo-cho, Chikusa-ku, Nagoya, Japan*

[3]*School of Engineering, University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan*

Keywords: Non-task-oriented, Dialogue Agent, Crowdsourcing, Gamification.

Abstract: This paper provides a novel method for building non-task-oriented dialogue agents such as chatbots. The dialogue agent constructed using our method automatically selects a suitable utterance depending on a context from a set of candidate utterances prepared in advance. To realize automatic utterance selection, we rank the candidate utterances in order of suitability by application of a machine learning algorithm. We employed both right and wrong dialogue data to learn relative suitability to rank the utterances. Additionally, we provide a low-cost and quality-assured learning data acquisition environment using crowdsourcing and gamification. The results of an experiment using learning data obtained via the environment demonstrate that the appropriate utterance is ranked on the top in 82.6% of cases and within the top 3 at 95.0% of cases. Results show that using context information that is not used in most existing agents is necessary for appropriate responses.

1 INTRODUCTION

A great demand exists for computerized dialogue agents. They are increasingly used in many different areas. Dialogue agents are categorizable into two types according to their task perspective: task-oriented dialogue agents and the non-task-oriented dialogue agents(Isomura et al., 2009). Task-oriented dialogue agents are used to accomplish particular tasks such as reservation services (Zue et al., 1994), supplying specific information (Chu-Carroll and Nickerson, 2000), etc. Non-task-oriented dialogue agents have no such tasks and only chat with us.

Non-task-oriented dialogues play a critical role in human society because they are an important tool for building relationships. Robots and other anthropomorphic agents are expected to participate increasingly in our daily lives. Therefore, much more investigation is needed on how non-task-oriented dialogue agents can be designed so that they can develop good relationships with people.

Even a task-oriented dialogue agent can accomplish a task more efficiently using non-task-oriented dialogues. For example, a study by Bickmore showed that when dialogue agents that supported the buying and selling of real estate initially chatted about subjects not pertinent to real estate such as the weather, people were much more motivated to buy real estate through them than through agents that did not engage in non-task-oriented dialogues (Bickmore and Cassell, 2001).

As described in this paper, we propose a construction method for non-task-oriented dialogue agents that are based on the statistical response method. In fact, two major response methods exist for non-task-oriented dialogue agents.

The first of these are rule-based methods that produce utterances in accordance with response rules. Well-known dialogue agents which use this strategy are ELIZA (Weizenbaum, 1966) and A.L.I.C.E. (Wallace, 2009). Mitsuku (Worswick, 2013) which the Loebner prize contest[1] (non-task-dialogue agent competition) winner of 2013 also used this strategy. The problem of this strategy is their substantial cost because the rules are developed by hand work.

The other is example-based method (Murao et al.,

[1]http://www.loebner.net/Prizef/loebner-prize.html

Inaba M., Iwata N., Toriumi F., Hirayama T., Enokibori Y., Takahashi K. and Mase K..
Constructing a Non-task-oriented Dialogue Agent using Statistical Response Method and Gamification.
DOI: 10.5220/0004722000140021
In *Proceedings of the 6th International Conference on Agents and Artificial Intelligence* (ICAART-2014), pages 14-21
ISBN: 978-989-758-015-4

2003; Banchs and Li, 2012). A dialogue agent employing this strategy searches a large database of dialogue by user input (a user's utterance) using cosine similarity and selects an utterance that follows after the most similar one as a response. The problem is how it acquires a large quantity of good quality dialogues efficiently because the performance depends on the quality of dialogues in the database. A response method based on statistical machine translation (Ritter et al., 2011) has been proposed. It treats last user's utterance as input sentence and translates it into the response utterance. This method is categorized as the example-based method and has same problem.

The mutual problem of the two is that it cannot use a context (sequence of utterances) but a given last user's utterance. According to the rule-based method, necessary rules and the costs of creating them are extremely increased. Regarding example-based methods, if it searches the database by a context, in many cases, then it cannot find a similar one because of the diversity of non-task-oriented dialogues. When the method cannot find a similar one, it has no choice but to use random selection.

Our statistical response method belongs to the category of example-based method because it uses dialogue data. However, our method, which uses no cosine similarity but statistical machine learning, is able to use contexts. Our method prepares candidate utterances in advance. It learns which utterances are suitable for context by the data. Therefore, a dialogue agent that is constructed using our method automatically selects a suitable utterance depending on a context from candidate utterances. Additionally, we provide a low-cost and quality assurance method of learning data acquisition using crowdsourcing and gamification.

2 STATISTICAL RESPONSE METHOD

2.1 Selection of Candidate Utterances

As described in this paper, we define "utterance" as a one-time statement and "context" as an ordered set consisting of utterances from the conversation's beginning to the specific point in time. Here, "an utterance is suitable to a context" means the utterance is a "humanly" and semantically appropriate answer to the context.

First, we define a state of a point of time in a dialogue as context $c = \{u_1, u_2, \ldots, u_l\}$. Each $u_i (i = 1, 2, \ldots, l)$ denotes an utterance appearing in the context and l denotes a number of utterances. Herein, u_1 is the last utterance; u_l is the first utterance in context c. As a matter of practical convenience, u_0 represents a response utterance to context c.

Table 1: Example of context c.

No.	Speaker	Utterance
u_4	Agent	Are you good at English?
u_3	Human	No I am not. I love Japanese.
u_2	Agent	It is said that experience is important to enhance English communication skills.
u_1	Human	I see! It might be a good idea to travel abroad during summer vacation.
u_0	(Agent)	(Select an utterance from Table2)

Table 2: Example of candidate utterance set A_c.

No.	Utterance
a_1^c	Are you good at English?
$a_2^c[r_1^c]$	Where do you want to go?
a_3^c	I think dogs are trustworthy and intelligent animals.
$a_4^c[r_2^c]$	That would be nice.
	...
$a_{20}^c[r_3^c]$	Travel can make a person richer inside.
	...
a_{130}^c	A link exists between mental and physical health.

Second, we define a candidate utterance set $A_c = \{a_1^c, a_2^c, \ldots, a_{|A_c|}^c\}$, where $a_i^c (i = 1, 2, \ldots, |A_c|)$ denotes a candidate utterance. Here, A_c contains suitable and unsuitable utterances to context c. $|A_c|$ represents a number of candidate utterances. We define the correct utterance set $R_c = \{r_1^c, r_2^c, \ldots, r_{|R_c|}^c\} \subseteq A_c$, where $r_i^c (i = 1, 2, \ldots, |R_c|)$ denotes a correct utterance. $|R_c|$ represents a number of correct utterances to context c. The utterance selection means acquiring a correct utterance set R_c from a candidate utterance set A_c, given a context c. Here, we assume that c and A_c fulfill the following requirements.

- A_c can be generated by any context c.
- A_c has at least one correct utterance r_i^c for context c.

Table 1 and 2 present examples of c, A_c, and R_c (R_c is shown by the darker-shaded area). In this example, a suitable utterance to context c shown in Table 1 is selected from the candidate utterance set A_c shown in Table 2. The utterance should be selected from the correct utterance set $R_c = \{a_2^c, a_4^c, a_{20}^c\}$ in this case.

2.2 Ranking Candidate Utterances

We describe the method used in our study to select candidate utterances automatically.

By specifically processing c and $a_i^c (\in A_c)$, we generate n-dimensional feature vector $\Phi(c, a_i^c) = \big(x_1(c, a_i^c), x_2(c, a_i^c), \ldots, x_n(c, a_i^c)\big)$ that represents relations between the context and the candidate utterance. Each $x_j(c, a_i)(j = 1, 2, \ldots, n)$ is a feature representing a binary value. For instance, when particularly addressing the last utterance u_1 in c and a_i, a feature $x_j(s, a_i^c)$ is represented if it contains a specific word, a word class, or a combination of the two.

We then defined f as a function that will return the evaluated value of a feature vector. In the following passages, we expressed the feature vector $\Phi(c, a_i)$ as Φ_i. Here it can be denoted using a linear function, which can be expressed as follows:

$$f(\Phi_j) = \sum_{j=1}^{n} w_j x_j(c, a_i^c). \tag{1}$$

Therein, w_j is a parameter representing the weight of $x_j(c, a_i^c)$

Using the evaluation function above, optimum utterance $\hat{a}$ in response to the context is obtainable by the following equation:

$$\hat{a} = \underset{a \in A_c}{\operatorname{argmax}} f(\Phi_j). \tag{2}$$

Therefore, the candidate utterances can be ranked by sorting the value from the above evaluation function.

To estimate the parameter $\boldsymbol{w} = (w_1, w_2, \ldots, w_n)$ in evaluation function f, we use a learning to rank method ListNet(Cao et al., 2007) algorithm.

2.3 Parameter Estimation

ListNet is constructed for ranking objects. It uses probability distributions for representing the ranking lists of objects. Then, minimizing the distance between learning data and distribution of the model, it learns suitable parameters for ranking.

We define $Y_c = \{y_1^c, y_2^c, \ldots, y_{|A_c|}^c\}$ as a score list to candidate utterance set $A_c = \{a_1^c, a_2^c, \ldots, a_{|A_c|}^c\}$. Each score $y_i^c (i = 1, 2, \ldots, |A_c|)$ denotes the score of a candidate utterance a_i^c with respect to context c. Score y_i^c represents the degree of correctness a_i^c to c and is an evaluated value given by humans. For instance, if a candidate utterance is a suitable response to a context, the score is 10. Alternatively, if an utterance is unsuitable, the score is 1.

ListNet parameter estimation algorithm uses pairs of $X_c = \big(\Phi_1, \Phi_2, \ldots, \Phi_{|A_c|}\big)$ which is a list of feature vectors and Y_c as learning data which are ranked correctly.

Here, for the list of feature vectors X_c, using function f, we obtain a list of scores $Z_c = \big(f(\Phi_1), f(\Phi_2), \ldots, f(\Phi_{|A_c|})\big)$. The objective of learning is to minimize difference between Y_c and Z_c in respect to their rankings. We then formalize it using a loss function.

$$G(C) = \sum_{\forall c \in C} L(Y_c, Z_c) \tag{3}$$

Therein, C means all contexts in learning data and L is loss function. In ListNet, the cross entropy is used as a loss function.

$$H(p, q) = -\sum_{x} p(x) \log q(x) \tag{4}$$

In that equation, $p(x)$ and $q(x)$ are probability distributions. When $p(x)$ and $q(x)$ show an equal distribution, cross entropy $H(p, q)$ takes a minimum value.

Therefore, the lists of scores Y_c and Z_c are converted into probability distributions using the Plackett–Luce model (Plackett, 1975; Luce, 1959). The distribution of Y_c using the Plackett–Luce model for the top rank utterance is expressed as follows.

$$P_{Y_c}(\Phi_i) = \frac{\mathrm{pow}(\alpha, y_i^c)}{\sum_{j=1}^{|A_c|} \mathrm{pow}(\alpha, y_j^c)} \tag{5}$$

In that equation, $\mathrm{pow}(\alpha, y)$ denotes α to the power of y. This equation represents the probability distribution of a candidate utterance being ranked on the top. The higher the candidate utterance score is, the higher the probability becomes. For instance, when a list of feature vectors X_c is (Φ_1, Φ_2, Φ_3) and a list of scores Y_c is $(1, 0, 3)$, then the probability of Φ_3 being ranked on the top is calculated as follows $(\alpha = 2)$.

$$\begin{aligned} P_{Y_c}(\Phi_3) &= \frac{\mathrm{pow}(2, y_3^c)}{\mathrm{pow}(2, y_1^c) + \mathrm{pow}(2, y_2^c) + \mathrm{pow}(2, y_3^c)} \\ &= \frac{\mathrm{pow}(2, 3)}{\mathrm{pow}(2, 1) + \mathrm{pow}(2, 0) + \mathrm{pow}(2, 3)} \\ &= 0.727 \end{aligned} \tag{6}$$

Instead, the probability of Φ_1 being ranked on the top is 0.182 and Φ_2 is 0.091, which is the lowest.

Similarly, the distribution of Z_c can be converted into a probability distribution as follows.

$$P_{Z_c}(\Phi_i) = \frac{\mathrm{pow}\big(\alpha, f(\Phi_i)\big)}{\sum_{j=1}^{|A_c|} \mathrm{pow}\big(\alpha, f(\Phi_j)\big)} \tag{7}$$

Using Eq. (4), (5) and (7), then the loss function $L(Y_c, Z_c)$ becomes

$$L\big(Y_c, Z_c\big) = -\sum_{i=1}^{|A_c|} P_{Y_c}(\Phi_i) \log\big(P_{Z_c(f)}(\Phi_i)\big) \tag{8}$$

Optimum parameter $\boldsymbol{\omega}$ is obtainable using Gradient Descent.

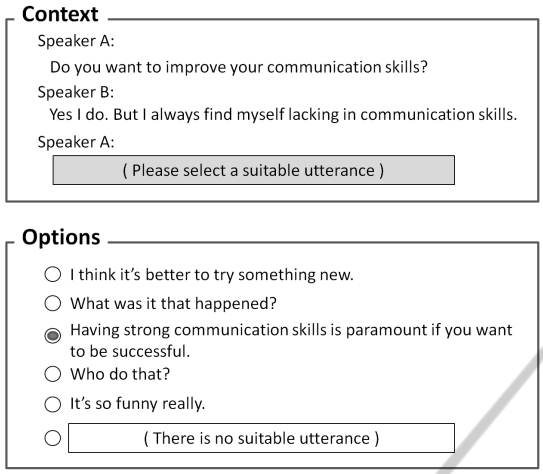

Figure 1: Context and candidate utterances on our crowdsourcing website.

3 DATA ACQUISITION

3.1 Crowdsourcing

Human work is important for acquiring data. To acquire data, we used crowdsourcing and opened a website for it.

The crowdsourcing website shows a context *c* and 5 candidate utterances (6 options) as shown in Figure 1 to participants. They select suitable candidate utterances to the context or "(There is no suitable utterance)". Then, we can acquire the pair of *c* and the selected utterance as a correct pair, or acquire *c* and these five utterances as an incorrect pair for learning. If a participant selects the option "Having strong communication skills is paramount if you want to be successful." as shown in Figure 1, then the pair of the context and the utterance are acquired as correct data.

When participants select "(There is no suitable utterance)", they must write a suitable utterance manually in the textbox. This way, we can acquire new candidate utterances. However, we do not use these utterances in crowdsourcing and subsequent experiments in this paper because they entail some problems such as spelling errors, phraseology, etc. We will use this function to collect new utterances continuously and produce a dialogue agent to handle even the newest topics in the future.

3.2 Confidence Estimation

When we use crowdsourcing, quality control of acquired data is necessary. We offer this to the general public. Therefore, quality gaps are unavoidable. In this study, we prepare several evaluated questions that comprise a context *c*, unsuitable utterances, and one or more suitable utterances. The suitability and unsuitability are judged in advance by four evaluators. We adopt the utterances which reach a consensus on the suitability or unsuitability among evaluators as evaluated ones.

The website measures the degree of confidence p by these evaluated questions. The degree of confidence p is calculated by counting how many times the suitable utterance is selected within N_p trials. Consequently, the range of p is $0 \leq p \leq N_p$.

Our crowdsourcing website presents 10 questions in a row: 5 questions for data acquisition and 5 questions for measuring the degree of confidence ($N_p = 5$). We decide whether the acquired data are available or not according to the degree of confidence p because, if p is small, then the possibility exists that the participant did not work seriously. To let participants answer seriously for all questions, the website does not tell participants which question is intended for data acquisition.

3.3 Gamification

One of the most important considerations with crowdsourcing is rewards to participants. If we set high rewards, then we can gather many participants and acquire much data. To construct a better non-task-oriented dialogue agent that can accommodate topics of many kinds, it is desirable to acquire new data continuously. Although the agent requires many new data, setting high rewards increases the cost of construction and the unserious users who don't address the task properly.

In this study, we bring game mechanics to data acquisition to gather participants with no rewards. Bringing game mechanics, participants enjoy the task like game play. Such a method brings game mechanics to accomplish an objective called "gamification"(Von Ahn and Dabbish, 2004; Deterding et al., 2011).

3.4 Gamified Data Acquisition Environment

We opened a website "The diagnosis game of dialogue skills" [2] (Japanese text only) as a gamified crowdsourcing data acquisition environment. At this site, participants answer 10 questions and finally obtain a score for dialogue skills. The score goes up to 100 points. The score becomes higher if a participant

[2] http://beta.cm.info.hiroshima-cu.ac.jp/DialogCheck/

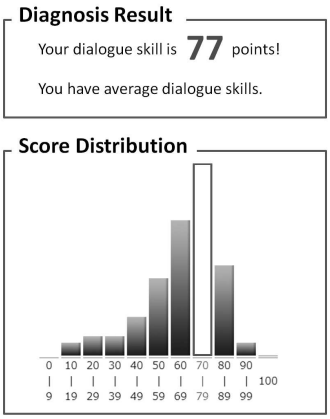

Figure 2: Diagnosis game result.

selects a candidate utterance that many other participants selected. At the same time, the website shows a graph of score distribution for comparison with other participants. Figure 2 portrays an example of a game result.

Scoring the results of selection and comparison with those of the other participants stimulates participants' retrial motivation, by which they want to obtain a higher score. Additionally, by posting the score on SNS or micro blogs by themselves, we expect advertising effects for other people (the website has a tweet button to tweet their score easily).

4 EXPERIMENTS

4.1 Experimental Methodology

To underscore the effectiveness of the statistical response method that learns data acquired through the gamified data acquisition environment, we checked the ranking of suitable utterances that were estimated automatically.

For comparison, we used a classification method, support vector machine (SVM). In general, SVM provides binary classification results and no direct means to obtain scores or probabilities for ranking. Nevertheless, Piatt proposed transforming SVM predictions to posterior probabilities by passing them through a sigmoid (Platt et al., 1999). We then classified candidate utterances by SVM, selected correctly classified ones and ranked them by posterior probabilities using the sigmoid method. We used this method as a baseline without the use of the learning to rank method.

4.2 Features

To rank the utterances, we converted pairs of a context and a candidate utterance into a feature vector. We used features of 11 types to represent relations between a context and an utterance. Here, we describe one of these, the noun feature, as the most basic one.

Table 3: Feature vector generation (noun feature).

No.	Speaker	Utterance
u_4	Agent	Enjoy this season fully because it's long-awaited summer vacation.
u_3	Human	Yes, I will.
u_2	Agent	Do you plan to travel?
u_1	Human	No. However, I would like to go.
u_0	(Agent)	Why don't you go on a trip overseas?

Noun pair	Vector value
u_1 : travel & u_0 : Europe	0
u_1 : summer & u_0 : trip	0
u_2 : part-timer & u_0 : overseas	0
u_2 : travel & u_0 : trip	1
u_2 : travel & u_0 : overseas	1
u_3 : friends & u_0 : trip	0
u_4 : summer & u_0 : overseas	1
u_4 : vacation & u_0 : trip	1

In the noun feature, we use a combination of a noun in a context and an utterance. Using this feature, we expect that a candidate utterance that includes words related to words in a context ranks higher. We only use u_1, u_2, u_3, and u_4 in a context for this feature because it is often the case that semantic relations between old utterances in a context and suitable candidate utterances are small. The usage range of utterances in context differs according to the type of feature. In the noun feature, whether a particular noun pair exists between utterances represents a binary feature value. We use noun pairs that appear three or more times in learning data.

Table 3 shows the example. The upper table shows an example of the context and candidate utterances. The lower shows part of a feature vector generated from them. As the table shows, we distinguish noun pairs by the number of utterances in the context. For instance, the vector value of "u_2 : travel & u_0 : overseas" is 1 because u_2 includes the word "travel" and u_0 includes "overseas". Similarly, the vector value of "u_1 : summer & u_0 : trip" is 0 because u_0 includes "trip" but u_1 does not include "summer".

The features should be designed to represent various aspects of relations between contexts and utterances, such as sentence structures, discourse structures, semantics, and topics.

Table 4: Data acquisition result.

Number of participants	460
Number of Evaluated contexts	320
Number of Evaluated utterances	4694
Average of the confidence p	4.215

4.3 Data Set

4.3.1 Candidate Utterances

We made 980 utterances by hand for crowdsourcing and the experiment. The topics of utterances were selected to interest as many people as possible such as healthcare, marriage, travel, sport, etc. We also produced versatile utterances such as "I think so." and "It's wonderful!".

4.3.2 Learning Data

To acquire the data, we opened the gamified website for crowdsourcing. Table 4 shows the results of data acquisition.

We used 4520 evaluated utterances for which confidence p is $p > 3.0$ for the experiment.

Additionally, we used other data produced by 50 part-time participants intended to compensate for data deficiency. The modes of producing data were about the same, with the exception of using the game mechanics. As a result, we obtained 239,897 evaluated utterances to 14,900 contexts. We used these data all together as learning data.

The scores of utterances are given depending on the evaluation. If an utterance is suitable to a context, then the score is 30. If unsuitable, the score is 1. The values of score are decided on an empirical basis.

4.3.3 Test Data

We prepared 500 contexts as test data. The ranked utterances using the proposed method and SVM were evaluated manually. Each utterance was evaluated by three evaluators. They judged whether each utterance was semantically suitable or unsuitable to the context. The eventual judge was decided using a majority. Therefore, when two evaluators judge an utterance as suitable and one evaluator judge as unsuitable, the utterance is determined to be suitable.

4.4 Results

Figure 3 shows the experiment result and 95% confidence interbals obtained using the proposed method and SVM.

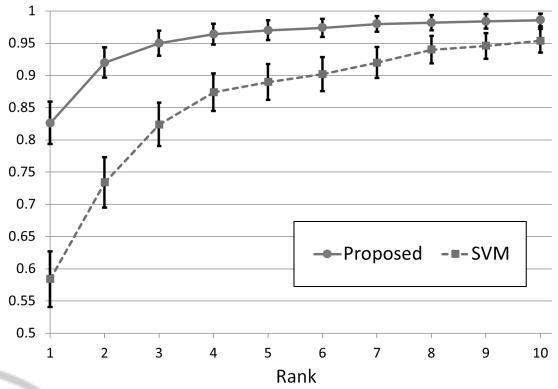

Figure 3: Rate of appropriate candidate utterance.

The x-axis represents the rank of the first appearance of a suitable utterance. The y-axis shows the cumulative frequency. In other words, the figure shows the rate of the contexts that include at least one appropriate utterance within each rank.

In the figure, the proposed method ranked a suitable utterance on the top at 82.6%, within the top 3 at 95.0%, and the top 10 at 98.6%. However, SVM was ranked on the top at 58.4%, within the top 3 at 82.4%, and at the top 10 at 95.4%. As shown in the result, the proposed method outperformed SVM overall. The above shows that the proposed method is effective for the selection of candidate utterances.

When we implement the proposed method to dialogue agents, the rate of replying to a suitable utterance (82.6%) is inadequate for smooth communication. Note that the set of candidate utterances has at least one correct utterance for each context (test data). This may not always be the case and the rate may drop when the agent talks to human actually. However, the proposed method produced rankings within the top 3 at over 90% to use new effective features. To improve the ranking algorithm, it seems that we can improve the performance of the statistical response method further.

4.5 Discussion

A great benefit of the proposed method is that it can use contexts for responses. To demonstrate that effectiveness, we created feature vectors using the last user's utterance (u_1) only and conducted an experiment.

Figure 4 portrays the results. The rate of the top 1 was 69.2%, 13.4% lower, and all results in the figure are lower by at least 1.6% than that using contexts (Fig. 3). This result is, so to say, the natural result because a context has more hints than an utterance for selecting a suitable utterance.

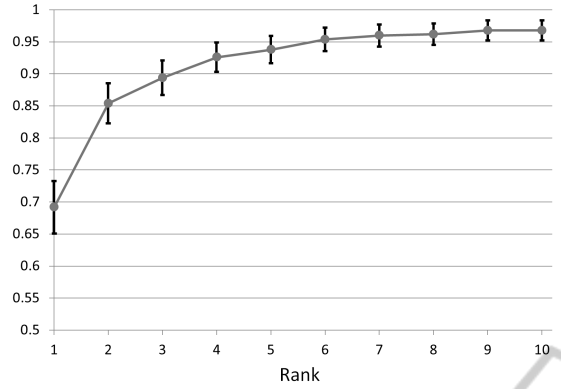

Figure 4: Rate of appropriate candidate utterance without use of contexts.

However, as described at the beginning of this paper, existing response methods cannot use contexts for response generation. Various problems exist because of such information loss. For instance, a dialogue agent broaches a topic that was discussed previously or makes contradictory comments to what it had said before. In fact, this experimentally obtained result indicates that using not only the last utterance but also contexts are necessary for realizing superior non-task-oriented dialogue agents. Therefore, in terms of the availability of contexts, the effectiveness of the statistical response method was clarified.

5 CONCLUSIONS

As described in this paper, we proposed a statistical response method that automatically ranks previously prepared candidate utterances in order of suitability to the context by application of a machine learning algorithm. Non-task-oriented dialogue agents that applied the method use the top utterance from the ranking result for carrying out their dialogues. To collect learning data for ranking, we used crowdsourcing and gamification. We opened a gamified crowdsourcing website and collected learning data through it. Thereby, we achieved low-cost and continuous learning data acquisition. To prove the performance of the proposed method, we checked the ranked utterances to contexts and conclude that the method is effective because a suitable utterance is ranked on the top at 82.6% and within the top 10 at 98.6%.

The non-task-oriented dialogue agents are basically evaluated by hand work and the task requires a tremendous amount of time and effort. By using proposed gamified crowdsourcing platform, we can evaluate the performance of non-task-oriented dialogue agents in a low-cost way. We prepare several types of agents which we want to evaluate and each agent generates a response to the given context. The platform shows the context and the generated responses to participants in the same way as our website. The responses which ware generated by a high-performance agent should be selected more than others.

The candidate utterances are created manually. Future work includes automatic candidate utterance generation. Our crowdsourcing website has a function that collects new utterances. However, these utterances present some problems such as spelling errors, phraseology, etc. because they are written by users in free description format. We need to fix them to use the new utterances. As an alternative utterance generation method, using microblog data is promising. Using microblog data, it can be expected to generate a new utterances set that includes numerous or newest topics.

We also intend to improve the feature vector. It is important to devise new effective features because the performance of our method depends heavily on the features. The features used in the experiment (not illustrated in detail here) did not deeply consider the semantics of contexts and utterances. Realizing appropriate responses requires semantical features. We are now deliberating on such features.

REFERENCES

Banchs, R. E. and Li, H. (2012). Iris: a chat-oriented dialogue system based on the vector space model. In *Proceedings of the ACL 2012 System Demonstrations*, pages 37–42. Association for Computational Linguistics.

Bickmore, T. and Cassell, J. (2001). Relational agents: a model and implementation of building user trust. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 396–403.

Cao, Z., Qin, T., Liu, T., Tsai, M., and Li, H. (2007). Learning to rank: from pairwise approach to listwise approach. In *Proceedings of the 24th international conference on Machine learning*, pages 129–136.

Chu-Carroll, J. and Nickerson, J. (2000). Evaluating automatic dialogue strategy adaptation for a spoken dialogue system. In *Proceedings of the 1st North American chapter of the Association for Computational Linguistics conference*, pages 202–209.

Deterding, S., Sicart, M., Nacke, L., O'Hara, K., and Dixon, D. (2011). Gamification. using game-design elements in non-gaming contexts. In *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems*, pages 2425–2428. ACM.

Isomura, N., Toriumi, F., and Ishii, K. (2009). Statistical Utterance Selection using Word Co-occurrence for a Dialogue Agent. *Lecture Notes in Computer Science*, 5925/2009:68–79.

Luce, R. (1959). *Individual choice behavior: A theoretical analysis*. New York: Wiley.

Murao, H., Kawaguchi, N., Matsubara, S., Yamaguchi, Y., and Inagaki, Y. (2003). Example-based spoken dialogue system using woz system log. In *SIGdial Workshop on Discourse and Dialogue*, pages 140–148.

Plackett, R. (1975). The analysis of permutations. *Applied Statistics*, pages 193–202.

Platt, J. et al. (1999). Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. *Advances in large margin classifiers*, 10(3):61–74.

Ritter, A., Cherry, C., and Dolan, W. B. (2011). Data-driven response generation in social media. In *Proceedings of the conference on empirical methods in natural language processing*, pages 583–593. Association for Computational Linguistics.

Von Ahn, L. and Dabbish, L. (2004). Labeling images with a computer game. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 319–326. ACM.

Wallace, R. (2009). The anatomy of alice. *Parsing the Turing Test*, pages 181–210.

Weizenbaum, J. (1966). ELIZA-a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1):36–45.

Worswick, S. (2013). Mitsuku Chatbot. http://www.mitsuku.com/.

Zue, V., Seneff, S., Polifroni, J., Phillips, M., Pao, C., Goodine, D., Goddeau, D., and Glass, J. (1994). Pegasus: A spoken dialogue interface for on-line air travel planning. *Speech Communication*, 15(3-4):331–340.