Lexical Acquisition through Implicit Confirmations over Multiple Dialogues

Kohei Ono[†], Ryu Takeda[†], Eric Nichols[‡], Mikio Nakano[‡] and Kazunori Komatani[†]
† The Institute of Scientific and Industrial Research (ISIR), Osaka University
Ibaraki, Osaka 567-0047, Japan
‡ Honda Research Institute Japan Co., Ltd.
Wako, Saitama 351-0188, Japan

Abstract

We address the problem of acquiring the ontological categories of unknown terms through implicit confirmation in dialogues. We develop an approach that makes implicit confirmation requests with an unknown term's predicted category. Our approach does not degrade user experience with repetitive explicit confirmations, but the system has difficulty determining if information in the confirmation request can be correctly acquired. To overcome this challenge, we propose a method for determining whether or not the predicted category is correct, which is included in an implicit confirmation request. Our method exploits multiple user responses to implicit confirmation requests containing the same ontological category. Experimental results revealed that the proposed method exhibited a higher precision rate for determining the correctly predicted categories than when only single user responses were considered.

1 Introduction

Much attention has recently been paid to *non-task-oriented* dialogue systems —or *chat-oriented* dialogue systems— both in research (Higashinaka et al., 2014; Yu et al., 2016) and in industry. In addition to pure chat-oriented systems, some task-oriented dialogue systems can engage in chat-oriented dialogues (Lee et al., 2009; Dingli and Scerri, 2013; Kobori et al., 2016; Papaioannou and Lemon, 2017) because such dialogues are expected to build *rapport* (Bickmore and Picard, 2005) between users and systems. For simplicity, we will call any system that can engage in chat-oriented dialogue a *chatbot*. Since an open-domain chatbot that always generates appropriate utterances is still difficult to build (Higashinaka et al., 2015), we think it is worth building a closed-domain chatbot, which tries to continue dialogues in a specific domain.

One problem in building closed-domain chatbots is that, although they should preferably have comprehensive lexical knowledge in their domains, all the knowledge cannot realistically be prepared in advance. Therefore, we must consider the case where a user uses terms outside of the system's vocabulary[1], i.e. terms that have ontological categories the system does not know. If the system can acquire the term's category during dialogues, it will be able to interact with users more naturally and the cost of expanding its knowledge base will be reduced.

We call the problem of acquiring the category of an unknown term *lexical acquisition*. If the system can predict the category of an unknown term, it can ask the user if it is correct (Otsuka et al., 2013; Komatani et al., 2016). However, repeating such explicit confirmation requests can degrade the user experience in chat-oriented dialogues[2]. We therefore need to find a way to enable chatbots to: (1) interact with the user naturally and (2) acquire lexical information. To solve this dilemma, we proposed an approach using *implicit confirmation* (Ono et al., 2016), where the system makes a confirmation request about the predicted category and uses the user's response to decide if the category is correct or not. However, whether such an approach is really possible or not has not been well studied.

This paper proposes a method that utilizes im-

[1] Here, we use *term* to mean an expression denoting an entity that can be in the knowledge base. A term may consist of multiple words.

[2] Some typical examples will be shown in Section 2. We will verify this intuition by conducting a user study.

Proceedings of the SIGDIAL 2017 Conference, pages 50–59,
Saarbrücken, Germany, 15-17 August 2017. ©2017 Association for Computational Linguistics

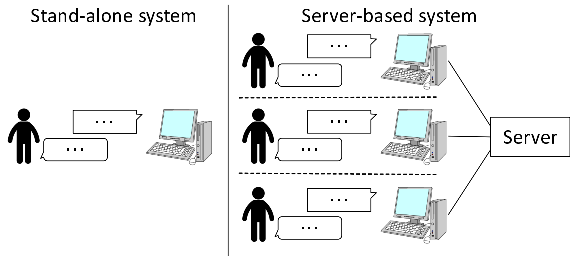


Figure 1: Server-based system can confirm the same prediction with different users

plicit confirmation dialogues from multiple users to increase the accuracy for determining if the predicted category is correct or not[3]. The system estimates the confidence score that the category prediction is correct from the responses of multiple users to the same implicit confirmation requests (Figure 1: right). Our proposed method has the goal of improving the confidence score estimation by using implicit confirmation sub-dialogues with multiple users. Then the system can determine if it should add the lexical information to the system's knowledge. For a sub-task, we consider the problem of estimating how likely the predicted category is to be correct from implicit confirmation sub-dialogues with one user (Figure 1: left).

It is reasonable to assume that the system can make confirmation requests about the same unknown term with different users because chatbots typically run on servers so they can share interaction logs for different users. Furthermore, it is difficult to ask a single user to respond to confirmation requests with the same predicted category many times, so collecting responses from multiple users is desirable.

This paper is organized as follows. The problem settings and related work are discussed in the next two sections. Section 4 describes the proposed method to determine correct categories in implicit confirmation requests on the basis of multiple implicit confirmation sub-dialogues with different users. Sections 5 and 6 show the data collection by crowdsourcing and several results as preparation for the main experimental evaluation of the proposed method, which is detailed in Section 7. Section 8 concludes this paper and discusses future work.

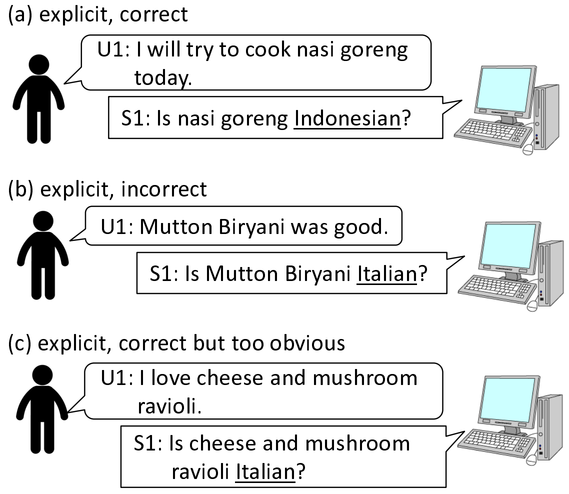

(a) explicit, correct
U1: I will try to cook nasi goreng today.
S1: Is nasi goreng Indonesian?
(b) explicit, incorrect
U1: Mutton Biryani was good.
S1: Is Mutton Biryani Italian?
(c) explicit, correct but too obvious
U1: I love cheese and mushroom ravioli.
S1: Is cheese and mushroom ravioli Italian?

Figure 2: Examples of explicit confirmation requests

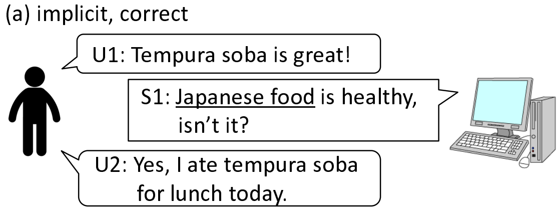
2 Problem Setting

This section describes the problem we address in this paper in detail. We are building a closed-domain Japanese language chatbot targeting the food and restaurant domain, so we use examples in this domain throughout this paper. In this domain, the problem is to acquire the categories of foods that the system does not know. We assume that the system can identify a food name in the user's input even if it is not in the system's vocabulary by using methods such as named entity recognition (Mesnil et al., 2015). Note that in this paper we also assume the category of an unknown term is predicted with an existing method (Otsuka et al., 2013; Ono et al., 2016). We do not assume any ontological structure of foods.

This paper focuses on deciding if the predicted category of unknown terms is correct or not in dialogues. To this end, methods for generating explicit confirmation have been proposed. Otsuka et al. (2013) proposed lexical acquisition methods that explicitly ask the user questions on the basis of category prediction results. For example, if the system does not know *nasi goreng* in the user input (denote as `U1`) in Figure 2 (a), the system predicts its category as *Indonesian food* and asks the user "Is nasi goreng Indonesian?"[4] Komatani et al. (2016) also proposed a utility-based method for selecting appropriate questions

[3] We do not deal with multi-party dialogues but utilize the interaction logs of two-party dialogues with different users.

[4] Note that Figures 2 through 4 show artificial examples, rather than those excerpted from the experimental data described in Section 5 because the experimental data are in Japanese and their direct translations are not natural.


(a) implicit, correct
U1: Tempura soba is great!
S1: Japanese food is healthy, isn't it?
U2: Yes, I ate tempura soba for lunch today.

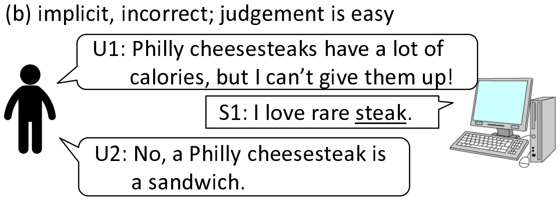

(b) implicit, incorrect; judgement is easy
U1: Philly cheesesteaks have a lot of calories, but I can't give them up!
S1: I love rare steak.
U2: No, a Philly cheesesteak is a sandwich.

Figure 3: Examples of implicit confirmation requests

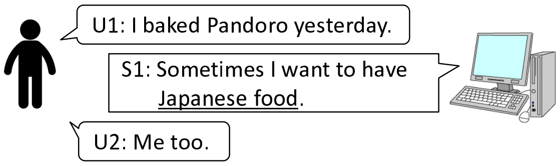

U1: I baked Pandoro yesterday.
S1: Sometimes I want to have Japanese food.
U2: Me too.

Figure 4: Example of implicit confirmation request for which judgement is difficult

on the basis of the results of category prediction. However, such explicit confirmation requests can degrade the user experience in chat-oriented dialogues, especially when the predicted category is incorrect as in Figure 2 (b), or the category of the unknown term is obvious as in Figure 2 (c).

We have proposed using implicit confirmation (Ono et al., 2016). For example, S1 in Figure 3 (a) does not explicitly ask the user if the category of *tempura soba* is Japanese, but from U2, it is possible to determine the category is correct. As another example, in Figure 3 (b), the system can determine the predicted category is incorrect from U2.

Determining if the predicted category is correct or not in implicit confirmation, however, is not always easy. Since user responses to implicit confirmation requests can come in various forms, looking at just the linguistic expressions of the user responses is not enough. For example, in Figure 4, the system incorrectly predicts the category *Japanese food* for *Pandoro* mentioned in U1 although it is Italian and generates an implicit confirmation request, S1. The user then talks about Japanese food to continue the dialogue (U2). In such cases, it is not simple to determine if the category is incorrect. If the system's determination is wrong, it might add incorrect information to its database. Thus, we need to find a way to accurately determine the correctness of the predicted categories through implicit confirmation.

3 Related Work

So far, several studies have addressed lexical acquisition in dialogues. Meng et al. (2004) and Takahashi et al. (2002) proposed methods for predicting the categories of unknown terms. They acquire coarse categories for unknown terms, which roughly correspond to named entity categories. Those categories can be acquired more easily than the more specific categories that we are trying to acquire. Holzapfel et al. (2008) proposed a method for a robot to acquire fine-grained categories for unknown terms by iteratively asking questions. We do not think this method is suitable for chatbots as it repeats explicit questions. Whereas a previous study tried to acquire relationships among domain-dependent entities in dialogues (Pappu and Rudnicky, 2014), here we focus on acquiring lexical information, which is required before such relations are obtained.

We address the problem of deciding if the content of an implicit confirmation request is correct or not. Some studies related to this problem have tried to classify affirmative and negative sentences by using rules or statistical methods. For example, de Marneffe et al. (2009) built rules for judging if a response to a yes/no question is affirmative or negative when it is not a simple "yes" or "no." Gokcen and de Marneffe (2015) investigated features for detecting disagreement in the corpus of arguments on the Web. In contrast, in this paper, we do not try to classify user responses into affirmative and negative ones but try to determine whether a category in an implicit confirmation request is correct or not. Furthermore, we utilize multiple sub-dialogues with different users.

Our method can be considered as an instance of implicitly supervised learning (Banerjee and Rudnicky, 2007; Komatani and Rudnicky, 2009) in that user responses to implicit confirmation requests are used as indicators for acquisition, though the target knowledge is different from those works.

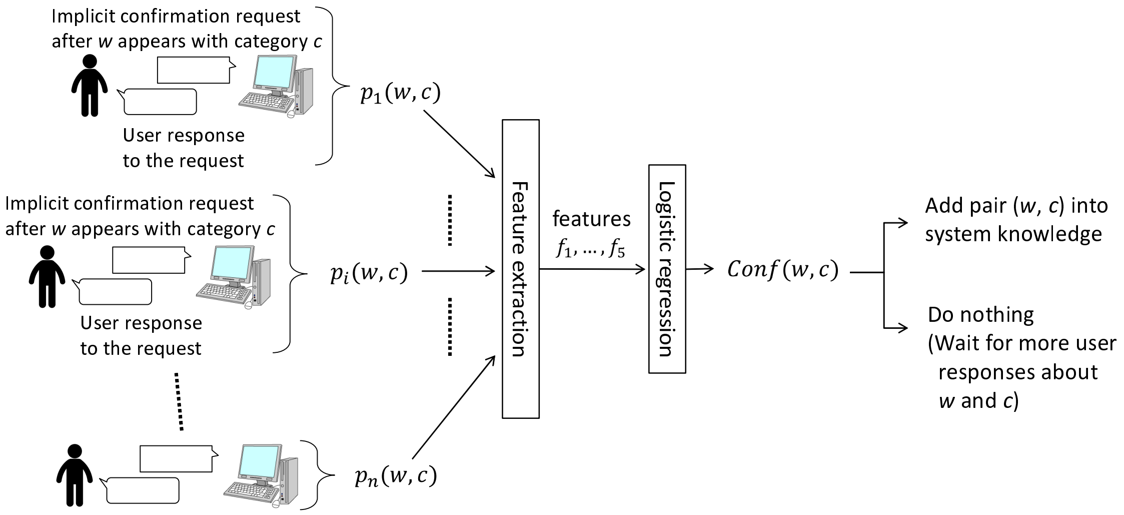

Implicit confirmation request after *w* appears with category *c*
User response to the request
$p_1(w,c)$
Implicit confirmation request after *w* appears with category *c*
User response to the request
$p_i(w,c)$
$p_n(w,c)$
Feature extraction
features $f_1,\dots,f_5$
Logistic regression
$Conf(w,c)$
Add pair (*w*, *c*) into system knowledge
Do nothing (Wait for more user responses about *w* and *c*)

Figure 5: Overview of calculating confidence score $Conf(w, c)$

4 Determining Correct Categories Using Responses from Multiple Users

The purpose of our method is to prevent the system from learning incorrect categories for an unknown term by using multiple implicit confirmation sub-dialogues with different users. This is possible because our system is designed as a server-based dialogue system and can give implicit confirmation requests with the same predicted category to different users. The proposed method determines more accurately whether or not the predicted category in the implicit confirmation request is correct by exploiting multiple responses to them.

Let $p_i(w, c)$ be the probability that a predicted category c of an unknown term w is correct after a single implicit confirmation request. The category can be predicted using surface information of the unknown term such as character n-gram and character types in Japanese (Otsuka et al., 2013). The index i denotes the i-th response to implicit confirmation requests. Our goal here is to obtain a confidence score $Conf(w, c)$ representing how likely category c of the unknown term w is to be correct on the basis of replies to implicit confirmation requests from n different users. We can then determine whether or not the system can add the pair of the unknown term w and category c into the system knowledge by setting a threshold for $Conf(w, c)$.

4.1 Procedure

Figure 5 gives an overview of the proposed method. The steps below initially start with $i = 1$.

1. Generate an implicit confirmation request containing a predicted category c for user i after an unknown term w appears.

2. Obtain the probability $p_i(w, c)$ from the implicit confirmation sub-dialogue with user i. The probability can be obtained by machine learning that has features based on expressions from the user response and its context.

3. Extract features from $p_1(w, c), ..., p_i(w, c)$ and calculate the confidence score $Conf(w, c)$ that represents how likely the category c of the unknown term w is to be correct.

4. If $Conf(w, c)$ exceeds a predetermined threshold, c is regarded as correct and is acquired as knowledge. Otherwise, increment i, go to Step 1, and generate one more implicit confirmation with c to another user after the unknown term w appears.

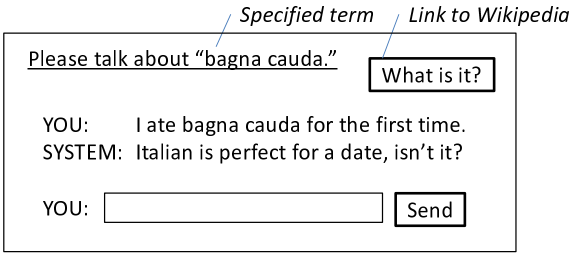
4.2 Obtaining Confidence Scores for Correct Categories

The problem of obtaining the confidence score $Conf(w, c)$ can be formulated as a regression using probabilities of n user responses $\{p_1(w, c), ..., p_n(w, c)\}$ as its input. Intuitively, the category c can be regarded as more likely to be correct when $p_i(w, c)$ with higher values are obtained more times.

Table 1 lists the features used in this regression for when probabilities $p_i(w, c)$ are obtained n times. To use the same regression function when

Table 1: Features from n responses $(1 \leq i \leq n)$

f1	Average of $p_i(w, c)$
f2	n
f3	$\max_i p_i(w, c)$
f4	$\min_i p_i(w, c)$
f5	$\lvert\{i \mid p_i(w, c) \geq 0.5\}\rvert / n$



Please talk about "bagna cauda."

YOU: I ate bagna cauda for the first time.
SYSTEM: Italian is perfect for a date, isn't it?

YOU:

Figure 6: Schematic diagram of GUI used in crowdsourcing

n increases, we design features that consist of a constant number even when n varies and that are derived from n responses to implicit confirmation requests with category c.

5 Data Collection via Crowdsourcing

We conducted experiments to verify if our method is effective. Although it would have been desirable to collect experimental data by incorporating our method into the chatbot we are developing and having it used by many people without giving any instructions, this would have required a huge amount of interactions to collect enough data to verify our method. We therefore collected user responses to implicit confirmation requests from 100 workers via crowdsourcing[5]. The data collection procedure consists of three steps: (1) a worker inputs an utterance containing a term specified on the interface at the crowdsourcing site, (2) the system generates an implicit confirmation request about the term, and (3) the worker fills in the response to the confirmation request. This procedure was repeated for 20 specified terms per worker.

Figure 6 shows a schematic diagram of the graphical user interface (GUI) used in the crowdsourcing. Note that it was actually in Japanese. The lines starting with "YOU" and "SYSTEM" denote the worker's and the system's utterances, respectively. At Step (1), the worker was asked to input an utterance that contains a term specified in the uppermost part in Figure 6. The worker was able to check the Wikipedia page for the specified term by following a link on the GUI. This was to prevent them from talking without understanding the term.

Table 2: Features for $p_i()$ with single user responses

g1	`U2` includes an expression affirmative to `S1`
g2	`U2` includes an expression negative to `S1`
g3	`U2` includes an expression correcting `S1`
g4	`U1` and `U2` contain the same word
g5	`U2` includes the category name used in `S1`
g6	`U2` includes a category name not used in `S1`, excluding cases that fall under g3
g7	`U2` includes a word preventing change of topic in `S1`
g8	`U1` includes the category name used in `S1`
g9	`U1` includes a category name not used in `S1`
g10	`U1` includes any interrogative
g11	`U1` includes an expression corresponding to the category mentioned in `S1`

We prepared 20 terms and their corresponding implicit confirmation requests used at Step (2): 10 had correct categories and the other 10 had incorrect categories. For example, for "shurasuko" (the Japanese rendering of churrasco), an implicit confirmation request with its correct category "meat dish[6]" is "Eating meat is fun, isn't it?" On the other hand, for "sangria," an implicit confirmation request with an incorrect category "yogashi[7]" is "Yogashi have a rich taste, don't they?" Furthermore, expressions of the implicit confirmation request were altered to make the confirmation request more natural when a worker's input was interrogative or negative.

We obtained 1,956 responses from 98 workers, half of which were responses to implicit confirmation requests with correct categories, and the other half were responses to those with incorrect ones. We removed data from two workers who just input only specified words or repeated the same sentences. We also removed four invalid inputs consisting of only spaces.

6 Preliminary Experiment with Single User Responses

6.1 Features for Obtaining Probabilities with Single User Responses

Table 2 lists the features for estimating how likely the categories in system confirmations are to be

[5] We used a crowdsourcing platform provided by Crowdworks, Inc. `https://crowdworks.co.jp/`

[6] Food category hierarchies usually used in Japan are different from those used in other countries.

[7] Yogashi means western sweets in Japanese.

correct. Here, U1, S1, and U2 respectively denote a user input, the implicit confirmation request by the system after U1, and the user response to the request. All feature values are binary; if the sentence for a feature is true, its value is 1, otherwise it is 0. These features were designed to represent differences in expressions of user responses to implicit confirmation requests with either a correct or incorrect category.

We briefly explain some important features by using the examples below. A user often uses affirmative expressions when responding to an implicit confirmation request with a correct category. This is represented by Feature g1, for which 15 affirmative expressions in Japanese were used such as "Yes" and "That's right."

When a category in an implicit confirmation request is correct, a user tends to continue with the same topic in U2 as in U1. In the example in Figure 3 (a), the user continues with the same topic and uses the same term *tempura soba* in U1 and U2. This is represented by Feature g4.

When the system makes an implicit confirmation request on the basis of an incorrect category, users tend to feel the system has suddenly changed the topic. In this case, the user tries to return the topic in U2 to the original one in U1. An example is as follows.

U1: I like sangria with its fruity taste.
S1: Yogashi have a rich taste, don't they?
U2: I am talking about the alcoholic beverage.

In this example, the system generates an implicit confirmation with the incorrect category "yogashi" in S1 although the correct category of sangria is "alcoholic beverage." Then the user says that the topic is an alcoholic beverage and tries to return to the original topic. Here, another category name not used in S1 is included in U2. This is represented as Feature g6.

For Feature g2, 17 negative expressions were used such as "is not [category name used in S1]" and "No." For Feature g3, six expressions such as "It is [category name not used in S1]" that tries to correct the system's previous confirmation request were used. Our system has 20 categories, and five more names such as "cheese" and "pasta" were used as category names for Features g6 and g9. Eighteen expressions including interrogatives were used for Feature g10.

Table 3: Confusion matrices with single responses

		Reference	
Features	Output	Correct	Incorrect
all	Correct	742	313
	Incorrect	236	665
g1, g2 only	Correct	320	220
	Incorrect	658	758

Table 4: Classification results with single responses

Features		P	R	F
all	Correct	0.703	0.759	0.730
	Incorrect	0.738	0.680	0.708
g1, g2 only	Correct	0.593	0.327	0.422
	Incorrect	0.535	0.775	0.633

P: precision, R: recall, F: F-measure

6.2 Classification Performance with Single User Responses

We conducted a preliminary experiment to classify responses to implicit confirmation requests with correct and incorrect categories. The data consists of the 1,956 responses and their contexts obtained by crowdsourcing as described in Section 5. We applied logistic regression to them with the features listed in Table 2. We used the module in Weka (version 3.8.1) (Hall et al., 2009) as its implementation. The parameters were the default values. The classification was performed by setting a threshold to the obtained probability $p_i(w, c)$. The threshold was 0.5, which is also the default value of Weka. Evaluation was conducted with a 10-fold cross validation.

We compared two feature sets: one consists of all 11 features listed in Table 2 and the other consists of Features g1 and g2 only. The latter corresponds to a baseline condition that only considers affirmative and negative expressions of U2 and does not consider any relationship with S1 and U1.

The results are shown in Tables 3 and 4. Table 3 shows confusion matrices of the raw outputs for the two feature sets. Table 4 summarizes the results as precision and recall rates and F-measures of the two categories (correct and incorrect) also for the two feature sets. The average-F scores, i.e. the arithmetic means of F-measures for the two categories, were 0.719 and 0.528 when all features and only g1 and g2 were used, respectively.

Table 5: Top-10 feature sets after removing arbitrary features for classification with single responses

Removed	Correct			Incorrect			
features	P	R	F	P	R	F	avg-F
g10	.704	.759	.730	.738	.681	.709	.719
None	.703	.759	.730	.738	.680	.708	.719
g7,g10	.701	.760	.729	.738	.676	.705	.717
g1,g4,g10	.699	.764	.730	.740	.672	.704	.717
g1,g4	.699	.765	.730	.740	.671	.704	.717
g7	.701	.759	.729	.737	.676	.705	.717
g4,g10	.691	.784	.735	.751	.649	.696	.715
g4	.690	.784	.734	.750	.648	.696	.715
g1,g4,g7,g10	.696	.765	.729	.739	.666	.700	.715
g1,g4,g7	.695	.766	.729	.739	.665	.700	.715

P: precision, R: recall, F: F-measure

This indicates that using the features representing context improves the classification more than using only the features obtained from U2.

We also performed feature selection to analyze which features were effective for the classification. More specifically, we performed the same experiments with all combinations of the 11 features, i.e., $2047 (= 2^{11} - 1)$ feature sets, and calculated their average-F scores. Table 5 lists top-10 feature sets sorted by the scores. "None" denotes the case when all the 11 features were used. First, the "None" condition was ranked second in the table, which shows that almost all features were effective for the classification. Next, when Feature g10 was removed, the F-value for the Incorrect category slightly improved and thus the average-F score also improved, as shown in the table. Because Feature g10 also appears in the table several times, Feature g10 was implied to be less helpful in this classification. On the other hand, the weight value for Feature g8 of the logistic regression function had the largest and positive value when Feature g10 was removed. This shows Feature g8 gave strong evidence and resulting $p_i(w, c)$ tended to be higher when Feature g8 was 1. This means that, when the common category name is included both in U1 and S1, the category included in S1 tended to be correct because the topic is not changed abruptly.

The results shown above indicate the classification performance was about 70% precision and recall rates on the basis of the user response and its context. However, we need higher precision because pairs of an unknown term and its predicted category will be added to the system knowledge, which must not contain errors. Thus, we have proposed a method using multiple user responses as described in Section 4, the effectiveness of which is verified in the following section.

7 Experimental Evaluation in Dialogues with Multiple Users

7.1 Data Preparation

In this section, we explain how to prepare data for training and evaluating the regression function to obtain $Conf(w, c)$. We performed the experiment in a perfectly open manner: no data were shared in training and test phases from the viewpoint of either workers or questions. More specifically, we had 98 (or 97) responses to implicit confirmation requests with 10 correct and 10 incorrect categories for making implicit confirmation requests, as explained in Section 5. Thus, we divided them into four disjointed groups, i.e., one group consists of 49 (or 48) workers with five correct and five incorrect categories.

The data were generated using responses collected from multiple users. The responses are mutually independent because they are obtained by a server-based dialogue system, so they can be combined in an arbitrary order. Thus, when we have N responses to single implicit confirmation requests, we can generate $\binom{N}{n}$ patterns. In our experiment, N was 49 (or 48) in each group. Since the values of $\binom{N}{n}$ become very large, we set a cut-off value when generating the combination randomly. The value was set to $1,000$ when $\binom{N}{n}$ exceeds $1,000$.

From this data combination, we obtained feature values listed in Table 1 with the reference values for every case. The reference value was set to either 1 or 0 depending on whether the category used in the implicit confirmation request was correct or not, respectively.

We then trained the regression function with each set of divided data of the four groups. We selected test data sets to be completely disjointed

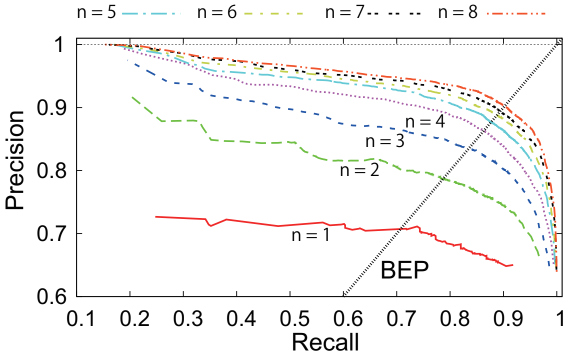


Figure 7: Precision and recall curves with BEP

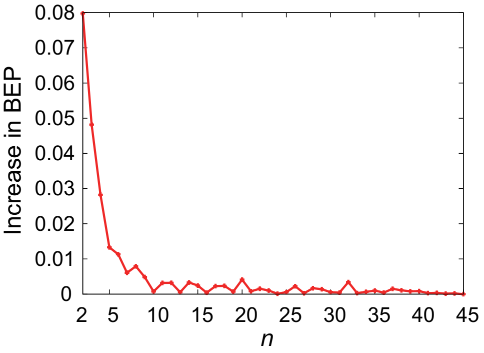


Figure 8: Increase in BEP values when n was incremented by 1

from each of the four data sets from the viewpoint of both workers and questions. We also used the logistic regression, which was implemented in Weka (version 3.8.1) (Hall et al., 2009), with its default parameters. The results by the regression for the four test sets are used together and analyzed hereafter.

7.2 Performance of Regression with Multiple Responses

We first investigated if the performance was better when the system used multiple responses from users. The precision and recall rates were calculated by setting various thresholds to $Conf(w, c)$ representing how likely a category c is to be correct for an unknown term w.

Figure 7 depicts the precision and recall curves for n up to 8. It also shows a line indicating the breakeven points (BEPs), meaning the value where the two rates are equal. The BEP is used as a single point representing a precision and recall curve and to show how good the estimated confidence score is when n changes. Note that $n = 1$ corresponds to the case when only single responses were used for the regression.

The performance represented by the BEP values became better as n became larger. In particular, the BEP values of $n \geq 2$ were larger than that of $n = 1$. This proves that the proposed method using multiple user responses more accurately determines whether the predicted category is correct or not.

We also performed feature selection by removing arbitrary features listed in Table 1. The performance of the regression function was measured by the summation of BEP values for each n $(1 \leq n \leq 48)$. The result revealed the best performance in the case was obtained when we used only Features f3 and f4. One reason for this result was that the correlations among the features might be high. We still need to further investigate feature sets to obtain better $Conf(w, c)$, which is future work.

7.3 Discussion on Reasonable Number of Responses

We discuss the relationship between the values of n and the performance of the regression function in more detail. Figure 7 shows that the performance represented by the BEP improved when n increased. On the other hand, cost will need to be incurred for increasing n, i.e., collecting responses from more human users. Thus, we investigate how much the performance of the regression function changed when n increased.

We first investigated how the BEP values increased in accordance with n values. Figure 8 depicts the increases in the BEP values when n was incremented by 1. It shows the increases were large while $n \leq 5$. This result indicates that it is worthwhile to ask more users implicit confirmation requests with predicted category c especially while n is small, to more accurately determine whether or not the category is correct. The figure also shows that the improvement mostly diminished, especially when $n \geq 10$. This indicates that the effect by asking implicit confirmation requests to more human users shows diminishing returns as n increases from the viewpoint of the performance represented by the BEP.

We furthermore investigated recall rates when thresholds were set to $Conf(w, c)$ so as to keep precision rates high. In our problem setting, high precision rates rather than high recall rates are re-

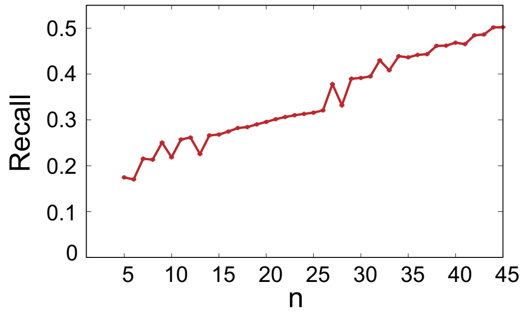


Figure 9: Recall rates with precision at 0.995

quired to avoid incorrect information being mistakenly added to the system knowledge. Figure 7 also shows the precision rate approached 1 for $n \geq 5$ by setting very large thresholds to $Conf(w, c)$. These cases indicate that the system can be almost perfectly confident that the predicted category c is correct. The recall rates were low for such cases because the precision and recall rates are in a trade-off relationship. We investigated the recall rates for such cases when n increased.

Figure 9 depicts the recall rates when we set very high threshold values for $Conf(w, c)$ so that the precision rates become almost one, i.e., $1 - \epsilon$. Here, we set $\epsilon = 0.005$[8]. First, the graph shows that the precision rate existed when n was 5 or more. For example, the recall rate for $n = 5$ was 0.175. This recall rate was rather low, but we think high precision rates should be prioritized over recall rates, even if some correct information is discarded at the current n. Second, the graph also shows that the recall rates increased with n. This means that, if the system asks more implicit confirmation requests with category c, more unknown terms the categories of which are c will be acquired with a sufficiently high precision rate.

8 Concluding Remarks

We have proposed a method to determine if the ontological category of an unknown term included in an implicit confirmation request is correct or not. Although responses to implicit confirmation requests seem to be insufficient for determining this, our method makes it effective by using the information on the context of the responses and exploiting responses from multiple users. Experimental results revealed that the proposed method exhibited higher performance than when only single user responses were used. We hope the performance will be improved with further feature engineering.

The proposed method is expected to enable a chatbot to acquire knowledge through dialogues without annoying users with repetitive simple explicit confirmation requests, while it can avoid acquiring wrong knowledge by achieving a high precision rate for determining the correctness of the knowledge.

We are planning to address several issues before deploying this method in a chatbot. Although we intuitively think implicit confirmation requests do not degrade users' impressions compared with repetitive explicit confirmation requests, we need to experimentally verify this by a user study. On the basis of its results, we will define a strategy of when to make implicit confirmation requests and when to make explicit confirmation requests. Despite these remaining issues, we believe that the experimental results presented in this paper are valuable in that they show the possibility of lexical acquisition through implicit confirmation.

Acknowledgments

This work was partly supported by JSPS KAKENHI Grant Number JP16H02869.

References

Satanjeev Banerjee and Alexander I. Rudnicky. 2007. Segmenting meetings into agenda items by extracting implicit supervision from human note-taking. In *Proc. International Conference on Intelligent User Interfaces (IUI)*. pages 151–159.

Timothy W. Bickmore and Rosalind W. Picard. 2005. Establishing and maintaining long-term human-computer relationships. *ACM Transactions on Computer-Human Interaction (TOCHI)* 12(2):293–327.

Marie-Catherine de Marneffe, Scott Grimm, and Christopher Potts. 2009. Not a simple yes or no: Uncertainty in indirect answers. In *Proc. Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*. pages 136–143.

Alexiei Dingli and Darren Scerri. 2013. Building a hybrid: Chatterbot – dialog system. In *Proc. International Conference on Text, Speech, and Dialogue (TSD)*. pages 145–152.

Ajda Gokcen and Marie-Catherine de Marneffe. 2015. I do not disagree: leveraging monolingual alignment

[8] The margin ϵ is required because the confidence score obtained by the logistic regression function cannot be 1 theoretically (the score can only converge to 1). Therefore, we selected the smallest ϵ with which we can calculate reasonable recall values.

to detect disagreement in dialogue. In *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*. pages 94–99.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA data mining software: an update. *SIGKDD Explor. Newsl.* 11:10–18.

Ryuichiro Higashinaka, Kotaro Funakoshi, Masahiro Araki, Hiroshi Tsukahara, Yuka Kobayashi, and Masahiro Mizukami. 2015. Towards taxonomy of errors in chat-oriented dialogue systems. In *Proc. Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*. pages 87–95.

Ryuichiro Higashinaka, Kenji Imamura, Toyomi Meguro, Chiaki Miyazaki, Nozomi Kobayashi, Hiroaki Sugiyama, Toru Hirano, Toshiro Makino, and Yoshihiro Matsuo. 2014. Towards an open-domain conversational system fully based on natural language processing. In *Proc. International Conference on Computational Linguistics (COLING)*. pages 928–939.

Hartwig Holzapfel, Daniel Neubig, and Alex Waibel. 2008. A dialogue approach to learning object descriptions and semantic categories. *Robotics and Autonomous Systems* 56(11):1004–1013.

Takahiro Kobori, Mikio Nakano, and Tomoaki Nakamura. 2016. Small talk improves user impressions of interview dialogue systems. In *Proc. Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*. pages 370–380.

Kazunori Komatani, Tsugumi Otsuka, Satoshi Sato, and Mikio Nakano. 2016. Question selection based on expected utility to acquire information through dialogue. In *Proc. International Workshop on Spoken Dialogue Systems (IWSDS)*. pages 27–38.

Kazunori Komatani and Alexander I. Rudnicky. 2009. Predicting barge-in utterance errors by using implicitly-supervised asr accuracy and barge-in rate per user. In *Proc. Annual Meeting of the Association for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL-IJCNLP)*. pages 89–92.

Cheongjae Lee, Sangkeun Jung, Seokhwan Kim, and Gary Geunbae Lee. 2009. Example-based dialog modeling for practical multi-domain dialog system. *Speech Communication* 51(5):466 – 484.

Helen Meng, P. C. Ching, Shuk Fong Chan, Yee Fong Wong, and Cheong Chat Chan. 2004. ISIS: An adaptive, trilingual conversational system with interleaving interaction and delegation dialogs. *ACM Transactions on Computer-Human Interaction (TOCHI)* 11(3):268–299.

Gregoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, and Geoffrey Zweig. 2015. Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23(3):530–539.

Kohei Ono, Ryu Takeda, Eric Nichols, Mikio Nakano, and Kazunori Komatani. 2016. Toward lexical acquisition during dialogues through implicit confirmation for closed-domain chatbots. In *Proc. of Second Workshop on Chatbots and Conversational Agent Technologies (WOCHAT)*.

Tsugumi Otsuka, Kazunori Komatani, Satoshi Sato, and Mikio Nakano. 2013. Generating more specific questions for acquiring attributes of unknown concepts from users. In *Proc. Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*. pages 70–77.

Ioannis Papaioannou and Oliver Lemon. 2017. Combining chat and task-based multimodal dialogue for more engaging HRI: A scalable method using reinforcement learning. In *Proc. ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. pages 365–366.

Aasish Pappu and Alexander I. Rudnicky. 2014. Learning situated knowledge bases through dialog. In *Proc. Annual Conference of the International Speech Communication Association (INTERSPEECH)*. pages 120–124.

Yasuhiro Takahashi, Kohji Dohsaka, and Kiyoaki Aikawa. 2002. An efficient dialogue control method using decision tree-based estimation of out-of-vocabulary word attributes. In *Proc. International Conference on Spoken Language Processing (ICSLP)*. pages 813–816.

Zhou Yu, Ziyu Xu, Alan W Black, and Alexander Rudnicky. 2016. Strategy and policy learning for non-task-oriented conversational systems. In *Proc. Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*. pages 404–412.