Developing Classifiers for Affirmative and Negative User Responses with Limited Target Domain Data for Dialogue System Development Tools

Yunosuke Kubo[1], Ryo Yanagimoto[1], Mikio Nakano[1,2], Kenta Yamamoto[1], Ryu Takeda[1], Kazunori Komatani[1]
[1]SANKEN, Osaka University, [2]C4A Research Institute, Inc.
Correspondence: komatani@sanken.osaka-u.ac.jp

Abstract

We aim to develop a library for classifying affirmative and negative user responses, intended for integration into a dialogue system development toolkit. Such a library is expected to highly perform even with minimal annotated target domain data, addressing the practical challenge of preparing large datasets for each target domain. This short paper compares several approaches under conditions where little or no annotated data is available in the target domain. One approach involves fine-tuning a pre-trained BERT model, while the other utilizes a GPT API for zero-shot or few-shot learning. Since these approaches differ in execution speed, development effort, and execution costs, in addition to performance, the results serve as a basis for discussing an appropriate configuration suited to specific requirements. Additionally, we have released the training data and the fine-tuned BERT model for Japanese affirmative/negative classification.

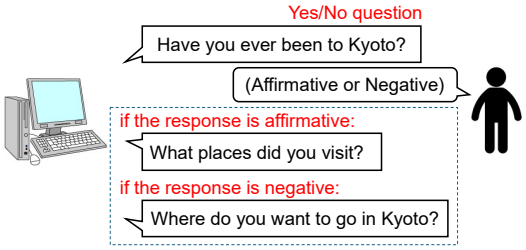

Figure 1: Example of dialogue flow changing on the basis of affirmative/negative classification.

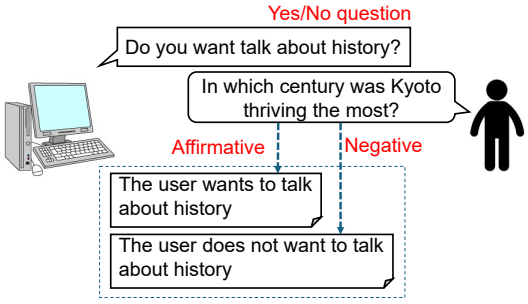

Figure 2: Example of acquiring user preferences on the basis of affirmative/negative classification.

1 Introduction

In dialogue systems, classifying whether a user's response to a system's question is affirmative or negative is a crucial and fundamental task. One reason for this is that the dialogue flow needs to be switched on the basis of the classification result (Figure 1). Although conducting dialogues using large language models (LLMs) has recently become possible, current systems may still struggle with progressing dialogues as intended by the system developers. The classification is also essential for deciding whether the system should retain the information, such as user preferences or factual knowledge, included in the system's question (Figure 2).

Users do not always respond to yes/no questions with simple expressions such as 'Yes' or 'That's right.' These types of responses are known as indirect answers (Louis et al., 2020). Furthermore, the user utterance shown in Figure 2, for example, is generally a question, but in a specific domain, it should be regarded as affirmative because it demonstrates interest in the topic. Such domain-specific classification also needs to be considered. Therefore, simple rule-based classification has its limitations, and utilizing machine learning, including LLMs, is a promising approach.

However, machine learning-based methods require training data. Collecting sufficient dialogue data and annotating it with correct labels is not practical for dialogue system developers who combine ready-for-use modules in toolkits. They prefer to minimize costs while still achieving a high-performance classifier.

We present experimental results on developing the affirmative/negative classifier when little or no annotated data is available in the target domain.

Proceedings of the 15th International Workshop on Spoken Dialogue Systems Technology, pages 309–317

The goal of this work is not to pursue higher performance but to discuss which configuration would be most appropriate for use in dialogue system development toolkits. We assume a situation where only a few dozen examples can be prepared by developers and used as target domain data. Several methods can be employed: fine-tuning a pre-trained BERT model (Devlin et al., 2019) with data from a different domain and the small amount of target domain data, and using a GPT API with few-shot learning by providing the target domain data as few-shot samples. Pre-trained models have become useful for several tasks, such as in extracting entity-value pairs for state tracking (Hudeček and Dusek, 2023; Bang et al., 2023).

In our experiments, we used dialogue data collected from real users during the finals of the Dialogue Robot Competition 2023 (Minato et al., 2024) as the test set. We incrementally added target domain data for fine-tuning or as few-shot samples and evaluated the classification performance. On the basis of the results, we discuss an appropriate configuration of the classifier, considering not only performance but also execution speed, development effort, and execution cost.

The contributions of this paper are as follows.

- We present experimental results to help determine appropriate configurations of an affirmative/negative classifier for dialogue system development toolkits.
- We have released the training data[1] and the general model for Japanese affirmative/negative classification[2].

2 Related Work

Several methods have been developed for classifying whether user responses are affirmative or negative. Asao et al. (2020) implemented a classifier using BERT, and Watanabe et al. (2023) developed the models using BERT and GPT. Such studies assumed large amounts of annotated data in the target domain. Several corpora containing indirect answers have also been collected and made publicly available (Louis et al., 2020; Damgaard et al., 2021; Sanagavarapu et al., 2022; Müller and Plank, 2024). In contrast, our goal is to provide an easily accessible classifier that can be integrated into dialogue system development toolkits, such as Rasa Open Source (Bocklisch et al., 2017) and DialBB (Nakano and Komatani, 2024). This paper shares the results of approaches that aimed at reducing the required effort by eliminating the need to collect and annotate large datasets.

Classifying whether an utterance is affirmative or negative can be considered a part of dialogue act classification (Stolcke et al., 2000), which has been addressed in various studies (Khanpour et al., 2016; Ahmadvand et al., 2019; Raheja and Tetreault, 2019). However, there are cases where an affirmative/negative classification cannot be made solely on the basis of the dialogue act. Often, there are expressions specific to the target domain, as shown in the example in Figure 2, and to situations where the preceding system utterance is a Yes/No question.

Phenomena such as indirect answers (Louis et al., 2020) have been theoretically examined from a linguistic perspective. Ginzburg et al. (2022) provided a taxonomy of responses to questions, while Enfield et al. (2018) analyzed responses to polar questions across 14 languages. Studies on dialogue management have also considered such responses (Larsson, 2002).

3 Experiment

Several methods were compared with minimal or no use of target domain data. The less annotated data required, the lower the cost for developers to build dialogue systems.

3.1 Task Formulation

Affirmative/negative classification is defined as follows: the input consists of a single exchange, specifically a pair comprising a Yes/No question from the system and the subsequent user response. The output is a three-class label: affirmative, negative, or other. The instructions for annotation, including details of the 'other' class, can be found in Appendix A.

3.2 Compared Methods

Five methods were considered:

(B0) Fine-tuning a BERT model with a large amount of data from a different domain.

(B1) Fine-tuning a BERT model with a small amount of target domain data.

[1] https://huggingface.co/datasets/ouktlab/Hazumi-AffNeg-Data

[2] https://huggingface.co/ouktlab/Hazumi-AffNeg-Classifier

(B2) Fine-tuning a BERT model with a large amount of data from a different domain and then further fine-tuning with a small amount of target domain data.

(L0) Using a GPT API with zero-shot learning.

(L1) Using a GPT API with few-shot learning.

The target domain data consist of exchanges (pairs of Yes/No questions and their responses) with annotated correct labels, similar to data from a different domain. (B0) and (L0) use no target domain data. For few-shot learning[3] in (L1), the target domain data are used in the prompts.

We chose BERT because publicly available pre-trained models are accessible. We also selected a GPT API for its generally high accuracy and ease of use but excluded fine-tuning of the GPT API due to its high costs. Performance comparison with other open-source LLMs will be left for future work.

3.3 Data

For the test data, we used dialogue data obtained from the finals of the dialogue robot competition (Minato et al., 2024). The domain is a tourist information task focused on Kyoto. These data were collected from 20 participants from the general public during the competition. From these conversations, we extracted 191 pairs of system Yes/No questions and the subsequent user responses. Consequently, 128 exchanges were labeled as affirmative, 56 as negative, and 7 as other.

For the target domain data, we used the dialogue data collected in our laboratory using a system (Yanagimoto et al., 2023) developed for the competition. Thus, the domain is the same as the test data. We extracted 131 pairs of the system's Yes/No questions and user responses. After two annotators labeled these pairs following the instructions in Appendix A, a high agreement rate of 0.977 was achieved. Therefore, one of the annotators labeled the remaining data, and the annotation results were considered the ground truth. The test data and those from a different domain were also annotated on the basis of the same criteria. The distribution of the 131 labels was 85 affirmative, 35 negative, and 3 other.

Note that, since we used speech recognition results as the user utterances in the target domain data, these may contain speech recognition errors and spelling mistakes. We did not manually correct these errors to better reflect real-world usage, as our goal is to develop a practical toolkit.

As data from a different domain, we used the Hazumi corpus (Komatani and Okada, 2021), which is a multimodal dialogue corpus recording chit-chat conversations between a human participant and a system. It initially includes 12 topics, but, in practice, centers around several topics such as food, games, and movies. Crucially, these topics differ from those in the target domain data and test data (tourist information in Kyoto). From the transcriptions (a total of 18,162 exchanges), we extracted 4,143 pairs of Yes/No questions and responses, labeled as follows: 2,864 affirmative, 1,017 negative, and 262 other.

3.4 Experimental Settings

As the BERT pre-trained model for (B0) to (B2), we used `tohoku-nlp/bert-base-japanese-v2`[4] and fine-tuned it using the JNLI script available at JGLUE[5](Kurihara et al., 2022) with the same parameters except for a batch size of 8 due to machine constraints.

In (B1) and (B2), the experiments were conducted using either a portion or all of the target domain data. Note that the topic of the target domain data is only tourist information in Kyoto. When using a portion, the partial data size varied from 10 up to 130, in increments of 10. The partial data were randomly selected from the target domain data, and this process was repeated five times. If the selected partial data did not contain all three-class labels, that subset was not used. We then calculated the average accuracy on the test data over the remaining subsets of each data size.

For (L0) and (L1), we used OpenAI's gpt-4o-2024-05-13[6]. We set the temperature parameter to 0.0 to obtain results as consistently as possible, and used the default values for the other parameters. An example prompt used in (L0) is shown in Appendix B. In (L1), similarly to in (B1) and (B2), experiments were conducted using either a portion or all of the target domain data. The partial data used were the same as in (B1) and (B2) and were incorporated into the prompt as few-shot examples. However, to reduce experimental costs,

[3] Since the amount of target domain data is relatively small, even when all of it is used, we refer to (L1) in this setting as few-shot learning.

[4] https://huggingface.co/tohoku-nlp/bert-base-japanese-v2

[5] https://github.com/yahoojapan/JGLUE

[6] https://openai.com/index/hello-gpt-4o/

Method	Accuracy
(B0)	0.817
(B1) with all target domain data	0.958
(B2) with all target domain data	0.942
(L0)	0.763
(L1) with all target domain data	0.949

Table 1: Accuracies with no target domain data and with all target domain data.

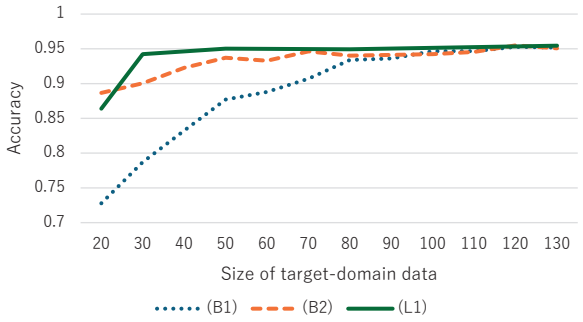

Figure 3: Accuracies when parts of target domain data were used in experimental settings (B1), (B2), and (L1).

only data sizes of 20, 30, 50, 80, and 130 were used, and each size was tested only three times, provided the randomly selected partial data contained the three class labels. Although the training data had imbalanced label distributions, we did not make adjustments, as it did not cause noticeable issues.

3.5 Results

Table 1 shows the results for (B0) and (L0), along with the results for (B1), (B2), and (L1) when all of the target domain data were used. More detailed results, including precision, recall, and F1 scores for the three labels, are provided in Table 3 in Appendix C. Neither (B0) nor (L0) achieved high accuracy. In contrast, when all target domain data were used, (B1), (B2), and (L1) eventually exhibited similar performance levels, which can be considered sufficiently accurate compared to the human annotation agreement rate.

Figure 3 displays the results for (L1), (B2), and (B1) as the amount of the target domain data increased. They performed better with smaller amounts of target domain data, in that order.

4 Error Analysis and Discussion

As shown in Figure 3, particularly while less target domain data were available, (B2) performed better than (B1). This was likely due to incorporating general patterns of affirmation and negation into the model through fine-tuning with the Hazumi

System: Do you have any concerns about Maruyama Park?
User: Can I walk there from Keage Incline?
(Correct) Affirmative; (Classified as) Other

System: Do you have any other questions?
User: Thank you, I'm fine.
(Correct) Negative; (Classified as) Affirmative

Figure 4: Examples of common errors in (B0)

data. (L1) performed even better with less data, presumably due to the extensive training data and the GPT model structure.

We examined the incorrect classification results in (B0) to investigate why the absence of target domain data results in low accuracy. A common pattern involved users responding to a system's questions with questions, as shown in Figure 4. Although users implicitly responded without 'yes' or 'no,' it appears that the different domain data, Hazumi, had too few instances of such patterns. These patterns also frequently resulted in errors in (L0). More detailed analysis with examples is provided in Appendix D.

Various surrounding factors in system development and operational circumstances (Nakano et al., 2024) should be considered when implementing and operating practical dialogue systems. There are several differences between pre-trained BERT models and GPT APIs beyond just accuracy, such as execution speed, development effort, and execution costs. Pre-trained BERT models are available for free, whereas GPT APIs require usage fees, making budget considerations necessary, and also require network connections, which can sometimes cause response delays. On the other hand, fine-tuning BERT can be time-consuming, and the resulting models are large, requiring considerable disk space and memory. In contrast, GPT APIs do not necessarily require training and impose less burden on the local machine. Therefore, the best approach should be chosen on the basis of the specific conditions.

5 Concluding Remarks

This paper presented an approach to developing an affirmative/negative response classifier using a small amount of target domain data. We have released the annotations for affirmative/negative classification on the Hazumi datasets, along with the classifier based on the BERT model, fine-tuned with this data.

The experimental results may be specific to the

particular dataset used. The performance depends on the model used, as well as the content of the data used for fine-tuning and few-shot learning. Nevertheless, we believe that the results and discussion could serve as a useful reference for developing dialogue systems, especially when working with little or no target domain data.

Future work includes conducting experiments using other datasets and models, and extending our approach to languages other than Japanese. Additionally, we will not only focus on binary affirmative/negative classification but also address the classification of agreement and disagreement (Gokcen and de Marneffe, 2015), which is crucial for dialogue management and knowledge acquisition (Komatani et al., 2022).

Limitations

In the experimental performance comparison presented in this paper, only a BERT model and a GPT API were used as representative models. Further evaluations using various other models are also necessary. The performance also depends on the content of the data used for fine-tuning and few-shot learning.

The experiments focused solely on the Japanese language, so conducting experiments in other languages, including English, remains a task for future work.

Acknowledgments

We thank the anonymous reviewer for their valuable comments on related research. This work was partly supported by JSPS KAKENHI Grant Number JP22H00536.

References

Ali Ahmadvand, Jason Ingyu Choi, and Eugene Agichtein. 2019. Contextual dialogue act classification for open-domain conversational agents. In *Proc. International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*, page 1273–1276.

Yoshihiko Asao, Julien Kloetzer, Junta Mizuno, Dai Saiki, Kazuma Kadowaki, and Kentaro Torisawa. 2020. Understanding user utterances in a dialog system for caregiving. In *Proc. International Conference on Language Resources and Evaluation (LREC)*, pages 653–661.

Namo Bang, Jeehyun Lee, and Myoung-Wan Koo. 2023. Task-optimized adapters for an end-to-end task-oriented dialogue system. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7355–7369.

Tom Bocklisch, Joey Faulkner, Nick Pawlowski, and Alan Nichol. 2017. Rasa: Open source language understanding and dialogue management. *Preprint*, arXiv:1712.05181.

Cathrine Damgaard, Paulina Toborek, Trine Eriksen, and Barbara Plank. 2021. "I'll be there for you": The one with understanding indirect answers. In *Proceedings of the 2nd Workshop on Computational Approaches to Discourse*, pages 1–11.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proc. North Americal Chapter of Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 4171–4186.

N. J. Enfield, Stivers Tanya, Brown Penelope, Englert Christina, Harjunpää Katariina, Hayashi Makoto, Heinemann Trine, Hoymann Gertie, Keisanen Tiina, Rauniomaa Mirka, Chase Wesley Raymond, Rossano Federico, Yoon Kyung-Eun, Zwitserlood Inge, and Stephen C. Levinson. 2018. Polar answers. *Journal of Linguistics*, 55(2):277–304.

Jonathan Ginzburg, Zulipiye Yusupujiang, Chuyuan Li, Kexin Ren, Aleksandra Kucharska, and Pawel Lupkowski. 2022. Characterizing the response space of questions: data and theory. *Dialogue Discourse*, 13(2):79–132.

Ajda Gokcen and Marie-Catherine de Marneffe. 2015. I do not disagree: leveraging monolingual alignment to detect disagreement in dialogue. In *Proc. Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 94–99.

Vojtěch Hudeček and Ondrej Dusek. 2023. Are large language models all you need for task-oriented dialogue? In *Proc. Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 216–228.

Hamed Khanpour, Nishitha Guntakandla, and Rodney Nielsen. 2016. Dialogue act classification in domain-independent conversations using a deep recurrent neural network. In *Proc. International Conference on Computational Linguistics (COLING)*, pages 2012–2021.

Kazunori Komatani and Shogo Okada. 2021. Multimodal human-agent dialogue corpus with annotations at utterance and dialogue levels. In *Proc. International Conference on Affective Computing and Intelligent Interaction (ACII)*, pages 1–8.

Kazunori Komatani, Kohei Ono, Ryu Takeda, Eric Nichols, and Mikio Nakano. 2022. User impressions of system questions to acquire lexical knowledge during dialogues. *Dialogue and Discourse*, 13(1):96–122.

Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. 2022. JGLUE: Japanese general language understanding evaluation. In *Proc. International Conference on Language Resources and Evaluation (LREC)*, pages 2957–2966.

Staffan Larsson. 2002. *Issue-based Dialogue Management*. Ph.D. thesis, Göteborg University.

Annie Louis, Dan Roth, and Filip Radlinski. 2020. "I'd rather just go to bed": Understanding indirect answers. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7411–7425.

Takashi Minato, Ryuichiro Higashinaka, Kurima Sakai, Tomo Funayama, Hiromitsu Nishizaki, and Takayuki Nagai. 2024. Overview of dialogue robot competition 2023. In *Proc. Dialogue Robot Competition 2023*.

Christin Müller and Barbara Plank. 2024. IndirectQA: Understanding indirect answers to implicit polar questions in French and Spanish. In *Proc. Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING)*, pages 9025–9035.

Mikio Nakano and Kazunori Komatani. 2024. DialBB: A dialogue system development framework as an educational material. In *Proc. Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 664–668.

Mikio Nakano, Hisahiro Mukai, Yoichi Matsuyama, and Kazunori Komatani. 2024. Evaluating dialogue systems from the system owners' perspectives. In *Proc. International Workshop on Spoken Dialogue System Technology (IWSDS)*.

Vipul Raheja and Joel Tetreault. 2019. Dialogue Act Classification with Context-Aware Self-Attention. In *Proc. North Americal Chapter of Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 3727–3733.

Krishna Sanagavarapu, Jathin Singaraju, Anusha Kakileti, Anirudh Kaza, Aaron Mathews, Helen Li, Nathan Brito, and Eduardo Blanco. 2022. Disentangling indirect answers to yes-no questions in real conversations. In *Proc. North Americal Chapter of Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 4677–4695.

Andreas Stolcke, Klaus Ries, Noah Coccaro, Elizabeth Shriberg, Rebecca Bates, Daniel Jurafsky, Paul Taylor, Rachel Martin, Carol Van Ess-Dykema, and Marie Meteer. 2000. Dialogue act modeling for automatic tagging and recognition of conversational speech. *Computational Linguistics*, 26(3):339–374.

Rikuto Watanabe, Junya Nakanishi, Jun Baba, Yuichiro Yoshikawa, and Hiroshi Ishiguro. 2023. Development of an affirmative/negative intention estimator for yes-no question answers using a large-scale language model (in Japanese). In *JSAI Technical Report, SIG-SLUD*, volume 98, pages 66–71.

Ryo Yanagimoto, Yunosuke Kubo, Miki Oshio, Mikio Nakano, Kenta Yamamoto, and Kazunori Komatani. 2023. User-adaptive tourist information dialogue system with yes/no classifier and sentiment estimator. In *Proc. Dialogue Robot Competition 2023*.

Label	Yes/No question	Response
1. Affirmative	Are there any places you recommend?	I recommend Okinawa.
1. Affirmative	Have you seen it?	I have seen it on TV.
2. Negative	Have you actually seen it?	I have only seen it on TV.
2. Negative	Do you plan to go there?	I will think about it.
3. Indeterminate	Do you like trains?	I think they are useful.
4. Does not answer	Are you interested in fashion?	What is ‘kasshon (misheard)’?

Table 2: Examples provided to the annotators

A Instructions for Annotators

We asked annotators to assign one of the following four labels to each exchange (a pair consisting of a system Yes/No question and the subsequent user utterance). Examples, such as those shown in Table 2, were also provided to them.

1. Affirmative
2. Negative
3. Indeterminate
4. Does not answer the question at all

The annotators were specifically instructed to focus on determining whether the response was essentially affirmative or negative, rather than relying on surface-level expressions, even if the response did not explicitly express either. Subsequently, Labels 3 and 4 were merged into ’other’ due to their small numbers.

B Prompt used in GPT-based classification

The zero-shot prompt used in (L0) is shown in Figure 5. The last question-response pair in the prompt was replaced with each of the test data to perform the evaluation across all test data.

On the basis of the human annotations, it outputs three values: affirmative, negative, and ‘noa,’ which refers to the ‘other’ class, including instances where the user does not respond to the question or where the response is undecidable.

In (L1), as part of a few-shot learning setup, we added pairs of dialogue examples and their correct labels from the target domain data to the prompt. The maximum number of such pairs was 131 in the conditions of (L1) with all target domain data.

C Further Details of Experimental Results

Table 3 presents more detailed results of Table 1. For each of the three labels, precision (P), recall (R), and F1 scores are shown. The upper part of each cell indicates the score, while the lower part shows the actual count. For (L0) and (L1), the counts represent the totals over three runs.

```
------------------------------------------------
Persons A and B had the following dialogue.
Please classify Person B's response to Person A's
question into one of the three categories below and
return it in JSON format (either {"class": "pos"},
{"class": "neg"}, or {"class": "noa"}).

pos: affirmative
neg: negative
noa: not answered

# input

A: Do you have any other questions?
B: That's OK. Thank you.
------------------------------------------------
```

Figure 5: Prompt used in GPT-based classification. (“noa” corresponds to the “other” class.)

Due to the small number of instances for the “other” label, its classification performance was unstable. “n/a” indicates that the percentage could not be calculated due to a zero denominator.

For the remaining two classes, the F1 scores followed the same trend as the accuracy results described in Section 3.5. Specifically, in order of highest to lowest accuracy—(B1), (L1), (B2), (B0), and (L0)—the macro-F1 scores for the Affirmative and Negative classes were 0.980, 0.977, 0.965, 0.898, and 0.878, respectively.

D Error Analysis When a Small Amount of Target Domain Data was Used

As shown in Figure 3, when the amount of the target domain data was very limited, the accuracy was higher in the order of (L1), (B2), and (B1). To investigate the reasons for this, we conducted an error analysis.

In the following, the experimental conditions for (B1), (B2), and (L1), where the number of the target domain data samples for training was 30, will be referred to as (B1-30), (B2-30), and (L1-30), respectively. The conditions where all training

		Affirmative			Negative			Other		
Method	Accuracy	P	R	F1	P	R	F1	P	R	F1
(B0)	0.817 156/191	0.936 102/109	0.797 102/128	0.861	0.962 51/53	0.911 51/56	0.936	0.103 3/29	0.429 3/7	0.167
(B1) with all target domain data	0.958 183/191	0.941 128/136	1.00 128/128	0.970	1.00 55/55	0.982 55/56	0.991	n/a 0/0	0 0/7	n/a
(B2) with all target domain data	0.942 180/191	0.927 127/137	0.992 127/128	0.958	1.00 53/53	0.946 53/56	0.972	0 0/1	0 0/7	n/a
(L0)	0.763 437/573	0.957 267/279	0.695 267/384	0.805	1.00 152/152	0.905 152/168	0.950	0.127 18/142	0.857 18/21	0.221
(L1) with all target domain data	0.949 544/573	0.986 365/370	0.951 365/384	0.968	1.00 163/163	0.970 163/168	0.985	0.400 16/40	0.762 16/21	0.525

Table 3: Detailed results corresponding to Table 1: precision (P), recall (R), and F1 scores for the three labels. Counts for (L0) and (L1) are totals over three runs.

data were used will be referred to as (B1-all), (B2-all), and (L1-all).

Note that classifications were made three times for (L0) and (L1-all), as GPT's classifications were not always the same even with the same prompt and a temperature setting of 0.0. If classification results vary within the same setting, all results are written in the following examples.

The following examples (1) and (2) illustrate the case where the classifications for (B1-30) were incorrect.

(1) System: I understand that you are visiting Kyoto this time, but do you travel often?
User: No, not really.

Correct: Negative
Classifications (B0): Negative
(B1-30): Affirmative
(B1-all): Negative
(B2-30): Negative
(B2-all): Negative
(L0): Negative
(L1-30): Negative
(L1-all): Negative

(2) System: Then, do you have any questions about this plan?
User: No, I'm fine.

Correct: Negative
Classifications (B0): Negative
(B1-30): Affirmative
(B1-all): Negative
(B2-30): Negative
(B2-all): Negative
(L0): Negative
(L1-30): Negative
(L1-all): Negative

We guess that (B1-30) failed because these patterns were relatively rare in the target domain data used for training, while the model trained with all the target domain data performed correctly. Since (B2-30) classified them correctly, it can be inferred that similar patterns existed in the different domain data, and using them was effective. We also guess that this is why the classifications of (B0), which is equivalent to (B2-0), where no target domain data were used in (B2), were correct.

The following examples (3) and (4) illustrate cases where both (B1-30) and (B2-30) were incorrect (twice in the three times), but (L1-30) was correct.

(3) System: I understand that you are visiting Kyoto this time, but do you travel often?
User: Would you go? (The intention of this response is unclear.)

Correct: Other
Classifications (B0): Other
(B1-30): Affirmative
(B1-all): Affirmative
(B2-30): Affirmative (twice), Other (once)
(B2-all): Affirmative
(L0): Other
(L1-30): Other
(L1-all): Other

(4) System: Then, are you interested in history?
User: History is... (incomplete response)

Correct: Other
Classifications (B0): Affirmative
(B1-30): Affirmative
(B1-all): Affirmative
(B2-30): Affirmative (twice), Other (once)
(B2-all): Affirmative
(L0): Other
(L1-30): Other
(L1-all): Other

In many of these cases, the correct label was 'Other.' Some user utterances made were difficult to classify as affirmative or negative due to misstatements, errors in detecting speech segments, or speech recognition errors. Since the target domain data for training included only three samples for the 'Other' class, the model did not have sufficient data to learn this classification effectively. GPT,

having been trained on a large amount of data, may also have had a higher chance of making the correct classification.

The following examples (5) and (6) illustrate cases where the classifications were incorrect in all conditions.

(5)	System:	In that case, do you have any questions about this plan?
	User:	*Home* (This is a misspelling of "ie," which means 'No' in Japanese.)
	Correct:	Negative
	Classifications:	(B0): Other (B1-30): Affirmative (B1-all): Affirmative (B2-30): Affirmative (B2-all): Affirmative (L0): Other (L1-30): Other (L1-all): Other (twice), Negative (once)
(6)	System:	Is there anything you are concerned about regarding the Keage Incline?
	User:	I would like to visit it.
	Correct:	Other
	Classifications:	(B0): Affirmative (B1-30): Affirmative (B1-all): Affirmative (B2-30): Affirmative (B2-all): Affirmative (L0): Affirmative (L1-30): Affirmative (L1-all): Affirmative

In Example (5), the speech recognition result was incorrectly transcribed, making it difficult for the models to accurately understand the user's utterance. In Example (6), because the user utterance does not explicitly address whether there are concerns, it can be reasonably classified as 'Other.' However, since the user mentioned wanting to visit the place due to a lack of concerns, the response might be interpreted differently, leading to potential annotation inconsistency. Such ambiguous situations can result in classification errors.