TransferTOD: A Generalizable Chinese Multi-Domain Task-Oriented Dialogue System with Transfer Capabilities

Anonymous ACL submission

Abstract

Task-oriented dialogue (TOD) systems aim to efficiently handle task-oriented conversations, including information gathering. How to utilize ToD accurately, efficiently and effectively for information gathering has always been a critical and challenging task. Recent studies have demonstrated that Large Language Models (LLMs) excel in dialogue, instruction generation, and reasoning, and can significantly enhance the performance of TOD through fine-tuning. However, current datasets primarily cater to user-led systems and are limited to predefined specific scenarios and slots, thereby necessitating improvements in the proactiveness, diversity, and capabilities of TOD. In this study, we present a detailed multi-domain task-oriented data construction process for conversations, and a Chinese dialogue dataset generated based on this process, **TransferTOD**, which authentically simulates human-machine dialogues in 30 popular life service scenarios. Leveraging this dataset, we trained a **TransferTOD-7B** model using full-parameter fine-tuning, showcasing notable abilities in slot filling and questioning. Our work has demonstrated its strong generalization capabilities in various downstream scenarios, significantly enhancing both data utilization efficiency and system performance.

1 Introduction

The Task-Oriented Dialogue System (TOD) is a human-computer interaction system aims to aid users in accomplishing specific tasks or acquiring particular information, which has found extensive use in daily life and commercial applications. At present, TOD systems have displayed the capability to adapt effectively to diverse tasks, domains, and user behaviors. Nonetheless, they continue to encounter various challenges related to generality, deep understanding, proactive questioning, and other aspects.

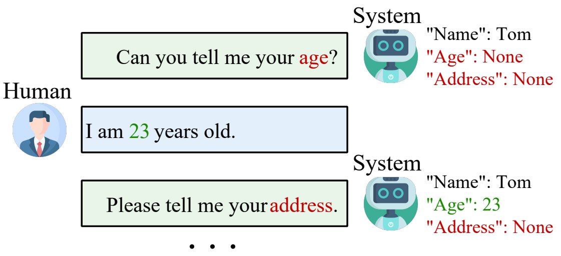

Figure 1: The system will ask the user for one or more slot values that are 'none', and then identify and update the corresponding field(s) based on the user's response until all slots are filled. For instance, if the system inquires about the user's age, and the user replies with '23', the model will update the slot form 'none' to 23.

To gather the necessary information, the system must proactively ask questions or guide users to provide the required information for filling specific slots, known as slot filling (SF) (Rosset et al., 2011). Traditional SF methods heavily rely on a substantial amount of annotated data (Fuisz et al., 2022), resulting in high manpower costs. To maximize data efficiency, domain experts have explored various approaches (Devlin et al., 2019; Liu et al., 2019; Henderson et al., 2020), such as transfer learning and fine-tuning. Recently, Large Language Models (LLMs) have exhibited promising performance in dialogue participation, instruction generation, and zero-shot reasoning (Zhang et al., 2023), owing to their remarkable dialogue capabilities. Research has confirmed that fine-tuned LLMs on dialogue corpora of different sizes can achieve enhanced performance across diverse tasks, domains, and even languages (Du et al., 2021; Touvron et al., 2023). TOD systems have commenced the simulation of human behavior (Park et al., 2023) and comprehension of user states (Feng et al., 2023; Su et al., 2023) through the Turing Test, significantly augmenting their proactiveness.

Nonetheless, Most methods in the field are limited to predetermined scenarios and specific slots. Certain methods even rely on external databases (Zhou et al., 2018; Tian et al., 2022; Zou et al., 2021), which restricts the system's ability to generalize across unknown domains or tasks. Furthermore, existing datasets primarily revolve around user-driven systems, where the focus is on constructing systems that primarily respond to user inquiries and requests (Wen et al., 2017; Budzianowski et al., 2020; Zhu et al., 2020). Addressing the diversity and authenticity of dialogue scenarios is also a crucial concern. The current data scenarios are often designed to be relatively narrow and rarely encompass the complex variations encountered in real conversations.

To address these issues, we introduce TransferTOD: a multi-domain, task-oriented Chinese dialogue dataset encompassing more complex and diverse dialogue tasks, and simulating more realistic conversation scenarios. Inspired by real-world questionnaire-style information-gathering scenarios, TransferTOD facilitates interactions between users and systems to assist in information acquisition and record updating. This dataset includes 35965 turns of statements and 5460 dialogues across 30 popular life service scenarios, providing researchers with a more challenging and practically significant dataset. Considering potential human errors and the variability of the Chinese context in practical applications, we have incorporated perturbed data and data polished through various methods into the dataset.

By selecting appropriate base models and fine-tuning methods, we have successfully demonstrated that training the TransferTOD-7B model using our dataset can achieve high accuracy. This model not only can proactively ask users for missing slots and accurately fill them based on their answers but also performs efficiently in guiding fluency and generating responses. Additionally, we evaluated the quality of the models in terms of slot filling ability and semantic accuracy in guiding user responses. The results indicate that our dataset can significantly improve model performance by handling noise, increasing question diversity, and optimizing language fluency.

Summarizing, the principal contributions of our paper are as follows:

1. We construct a new dataset called TransferTOD for task-oriented dialogue generation in various lifestyle service scenarios. It consists of 30 scenarios with 5460 dialogues, and ablation experiments have demonstrated that this dataset exhibits good noise resistance, diversity, and fluency.

2. We present a comprehensive dataset construction pipeline with high generalizability and transferability, enabling fellow researchers to effectively apply the methodology for creating datasets across various languages or in multilingual contexts.

3. We have utilized TransferTOD as our SFT dataset and trained the TransferTOD-7B model through full-parameter fine-tuning, achieving better slot filling and questioning capabilities comparable to GPT-4. Additionally, with appropriate secondary fine-tuning techniques, our model demonstrates superior performance in out-of-domain testing compared to GPT-3.5-Turbo fine-tuned with an equivalent amount of data.

2 Related Work

Task-oriented Dialogue Datasets The performance of intelligent dialogue systems is profoundly influenced by the quality of the dialogue datasets, making dataset construction an active research area. Initial generations of task-oriented dialogue datasets often focused on a single task or even a single scenario, ATIS (Hemphill et al., 1990), DSTC2 (Henderson et al., 2014), WOZ2.0 (Wen et al., 2017), etc. included. The emergence of these databases not only enhanced the conversational fluency of conversational agents but also made task completion through natural dialogues between machines and humans possible. Considering that user dialogues often involve domain transitions, datasets Multi-WoZ (Budzianowski et al., 2020), CrossWoZ (Zhu et al., 2020) etc. encompassing more scenes and larger volumes of data were subsequently proposed. However, these dialogues are user-led discussions on relevant topics, requiring a user to pose questions or set tasks for the dialogue agent to respond accordingly.

TOD System Enhancement Methodology Enhancing the performance and data utilization of TOD systems and strengthening their ability to understand specific tasks expressed by users remain hot research topics. To complete tasks and improve accuracy, (Li et al., 2018) proposed an end-to-end neural dialogue system based on reinforcement learning. TOD gradually started to realize across tasks (Peng et al., 2017), domains (Hakkani-Tür et al., 2016), and even languages (Wang et al.,

	Train	ID Test	OOD Test
# Domain	27	27	3
# Slot	188	188	27
# Dialogue	4320	540	600
# Turns	28680	3585	3700
# Slots / Dialogue	10.3	10.3	9.7
# Tokens / Turn	66.4	66.4	76.8

Table 1: Overall statistics of TransferTOD. ID Test means In-Domain test and OOD Test means Out-of-Domain test. The domains of the test set are Water-Delivery, Sanitation, and Courier.

2021). TOD-BERT (Wu et al., 2020), MinTL (Lin et al., 2020), Soloist (Peng et al., 2021), etc. has been successively proposed improving the success rate of tasks. However, as task complexity increases, these methods still rely heavily on large-scale datasets and lack competitiveness in handling noise robustness.

LLM-based TOD System Existing research (Brown et al., 2020; Chowdhery et al., 2022; Chen et al., 2021; OpenAI et al., 2023) has demonstrated LLMs' exceptional capabilities in natural language understanding, zero-shot reasoning, and command generation. With their advent and deep utilization, dialogue systems have entered the LLM-based era (Wang et al., 2023). Utilizing LLMs, many dialogue tasks have achieved significant breakthroughs. On one hand, through internal dialogues with users, systems can be equipped with human-like perception and reasoning abilities, including intent classification, semantic parsing, dialogue state tracking, and reply generation. On the other hand, the integration of external information sources, such as specific databases, memory knowledge sources, the internet, etc., ensures the system provides the latest, rich, accurate, personalized, and necessary information to complete tasks.

3 TransferTOD

3.1 DataSet

TransferTOD aims to construct a cross-disciplinary task-oriented information collection multi-turn dialogue dataset, encompassing tasks such as goal-oriented questioning, dialogue state maintenance, information collection, and parsing. Existing task-oriented Wizard of Oz (WoZ) datasets are typically user-driven systems with relatively single domains. Departing from scenarios in the real world where questionnaire-style information collection may occur, we have curated dialogues spanning 30 different domains. We have enhanced the data in terms of robustness, diversity, and fluency, ensuring that the data closely mirrors real-world situations.

Figure 2 illustrates the 4 steps of data collection and processing: 1. Original slot construction and dialogue generation; 2. Introduction of perturbed data; 3. GPT-enhanced dialogue diversity; 4. Manual refinement of dialogue content for fluency. Overall statistics of TransferTOD are shown in Table 1.

3.1.1 Field Selection and Slot Collection

We crawl the most popular 30 life service offerings from local lifestyle applications (such as Yelp) to construct the domain for our dialogue system. Specifically, we analyzed the submitted forms of each service, abstracting the information that the system would require users to provide as slots.

After constructing the slots, we built a corpus containing all possible values for each slot. For string-type slots, we adopted a method of collecting publicly available information from the internet and generating rules. During the collection process, we kept the information for each slot separate, ensuring that no real personal information was involved. For number-type data, we described its range and distribution, generating it in real-time during the dialogue construction process.

Human experts[1] manually created a set of high-quality dialogues as test data across 30 domains; three of these domains were selected for constructing an out-of-domain test set due to their minimal overlap in slots with the other domains. The remaining data is used as the in-domain test set. For the training dataset, the following steps will be undertaken to generate it on a large scale.

3.1.2 Dialog Construction

Based on existing slot type descriptions and vocabularies, we have implemented the first version of a dialogue dataset using a script-generated approach. Specifically, we constructed a template library for each domain. Each dialogue round consists of a user response, a system question, or a summary, forming the values before and after the dialogue state changes.

For the number of slots k that could potentially be extracted in a single dialogue, we experimented

---

[1] Details of the human experts will be shown in the appendix D.1

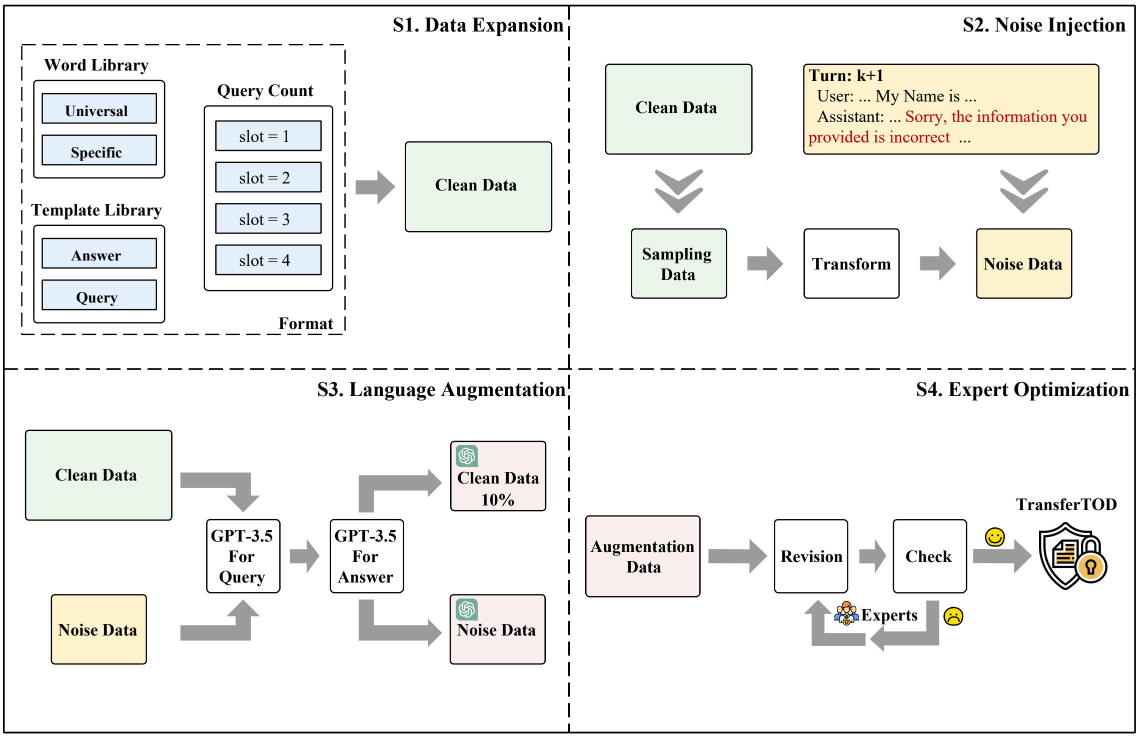

Figure 2: Our dataset development comprises four phases. Initially, we create specific scenarios, develop the corresponding questions and answers, and generate data for slots 1 to 4 using rule-based methods. In the second phase, we introduce noise into a subset of this data to simulate inaccuracies needing correction by customer service, prompting a re-query in the next interaction. The third phase diversifies the dataset by rephrasing both clean and noise data via GPT-3.5. In the final stage, expert professionals refine the input to achieve a high level of naturalness in customer service interactions, ensuring that the inquiries exhibit a seamless and fluent conversational flow.

with four scenarios: $k = 1, 2, 3, 4$. Specifically, when $k = 3$, the system simultaneously asks the user for information on 3 slots, and the user needs to respond to these three corresponding aspects. The statistical information of the original dialogue data is detailed in Table 1. The dataset obtained after this step is **TransferTOD-v1**.

3.1.3 Noisy Data Construction

In real-world scenarios, users may provide information that does not conform to standards or common sense. Therefore, a comprehensive dialogue system should possess the capability to scrutinize the responses provided by users and, when necessary, seek clarification to obtain accurate information. To address this, a portion of the data is delineated to incorporate rounds of interaction specifically designed to handle incorrect responses from users.

There are two types of data disturbances: 1. Non-responsive answers, where the content of the user's reply significantly deviates from the system's query. This dialogue alteration is achieved by replacing the user's response with an irrelevant answer; 2. Illogical responses, where the user's reply may contradict basic common sense. This data segment necessitates the introduction of non-factual content into the slot value lexicon to accommodate such instances.

During rounds with erroneous responses, the system will identify the user's mistake, repeat the original question, and maintain the dialogue state without updating it. We constructed 3013 noise dialogue data, with each dialogue containing at least one of the aforementioned errors, where the first type of error represented more than 90% of the cases. The dataset obtained after this step is **TransferTOD-v2**.

3.1.4 Dialogue Diversity and Fluency Polish

Dialogue data generated by static script schemes exhibit a shortfall in the diversity of questioning and answering modes. Each slot is confined to merely 5-6 variations of queries and responses, which fails to mirror the spectrum of linguistic preferences encountered in real-life scenarios. Consequently, we have leveraged the GPT-3.5 model to reformulate the texts of questions and answers, ensuring fidelity to the original intents while adjusting the temperature coefficient to 0.5 for an enriched array of textual content. The dataset obtained after this step is **TransferTOD-v3**.

Furthermore, we have refined the fluidity of dialogues that encompass inquiries about multiple slots within a singular exchange. Initially, dialogue data were essentially composed of simplistic amalgamations of disjointed questions or responses, not aligning with conventional spoken habits. Through the application of GPT-4 for sentence amalgamation and enhancement of coherence, coupled with rule-based scrutiny to pinpoint instances of fragmented sentences, we have engaged human annotators for the revision of overlooked or non-compliant sentences, thereby assuring dialogue smoothness. The dataset obtained after this step is **TransferTOD-v4**, which is our final dataset.

3.2 Models

Upon acquiring the TransferTOD dataset, we opted for the Baichuan2-7B-Base (Baichuan, 2023) as the foundational model for fine-tuning. During the model training process, we employed two methods: full-parameter fine-tuning (Zeng et al., 2023) and LoRA (Low-Rank Adaptation) fine-tuning (Hu et al., 2021).

3.2.1 Supervised Fine-tuning

To equip the model with basic conversational abilities, we initially combined the training subset of the TransferTOD dataset with the general Chinese conversational dataset BELLE (Ji et al., 2023) in equal proportions to construct the SFT (Supervised Fine-Tuning) dataset. This dataset was utilized for full-parameter fine-tuning of the Baichuan2-7B-Base model to derive the TransferTOD-7B model.

3.2.2 Secondary Fine-tuning

Following the development of our TransferTOD-7B model, we aimed for our model to achieve commendable performance in specific downstream tasks, necessitating that our model possesses superior generalization capabilities. In three external domain test sets, we adopted a limited-sample secondary fine-tuning approach to further enhance the accuracy of TransferTOD-7B in external domain test sets. Research (Sun et al., 2023) indicates that compared to full-parameter fine-tuning, LoRA fine-tuning achieves better generalization. Consequently, we employed LoRA fine-tuning for secondary fine-tuning. Experimental evidence demonstrates the effectiveness of our methodology in scenarios where data availability for downstream tasks is significantly constrained.

4 Experiment

In this section, we detail the experiments conducted. The primary experiments were carried out on the test set of TransferTOD. Additionally, we conducted ablation studies on the dataset construction phase, as well as supplementary experiments to further investigate the effects of secondary fine-tuning.

4.1 Experimental Setup

For the in-domain test, we evaluated various methods known for their effectiveness in slot extraction. Traditional TOD systems divide the task into several modules (Zhu et al., 2020), each managed by a distinct model, forming a system pipeline. However, LLMs can reduce the reliance on task decomposition, thereby allowing us to directly evaluate the core competency of slot filling through information extraction.

For the out-of-domain test, a model's ability to adapt and generalize is paramount. Consequently, our initial evaluation centered on a selection of open-source LLMs with parameter counts comparable to our base model (7 billion), all of which demonstrated strong performance in Chinese benchmarks. To further enhance our analysis, we incorporated two powerful, near-source models from OpenAI.

4.1.1 Baseline

For the in-domain test, we select 4 models as baseline: BertNLU (Zhu et al., 2020), SoftLexicon(LSTM) (Ruotian et al., 2020), LEBERT+CRF (Liu et al., 2021) and W2NER (Li et al., 2022).

For the out-of-domain test, we select 6 Large Language Models as baseline: Baichuan2 (Baichuan, 2023), ChatGLM3 (Du et al., 2022; Zeng et al., 2022), Qwen (Bai et al., 2023), Yi[2], GPT-3.5-Turbo[3], GPT-4 (OpenAI et al., 2023). Please refer to the appendix A.1 for details.

4.1.2 Implementation Detail

For evaluating the slot filling capability, we have annotated user utterances with BIO tags and trained 4 models for the in-domain test. A detailed system

---

[2] `https://github.com/01-ai/Yi`

[3] `https://platform.openai.com/docs/models/gpt-3-5`

Model	Dialogue Act F1(%)
BertNLU (Zhu et al., 2020)	79.32
SoftLexicon(LSTM) (Ruotian et al., 2020)	77.12
LEBERT+CRF (Liu et al., 2021)	79.72
W2NER (Li et al., 2022)	78.45
TransferTOD-7B	**93.64**

Table 2: Results of the in-domain test: The dialogue act F1 Score of each model, showing the accuracy of predicting the right dialogue acts from user utterance.

prompt was designed when inferencing with those LLMs in out-of-domain test. Please refer to the appendix A.2 for details.

4.1.3 Evaluation Metrics

For the out-of-domain test, we assess the model's capabilities in two main aspects: slot filling ability and semantic accuracy during the phase of guiding user responses. To evaluate the slot filling ability, we employ F1 and Joint Accuracy, which are widely used in the TOD systems for slot extraction tasks. To evaluate the semantic accuracy of model-generated questions, we use a manual evaluation approach. Please refer to the appendix A.3 for details.

4.2 Results on TransferTOD

This section shows the results of our main experiment.

4.2.1 Results on In-Domain Test

Table 2 presents the results of the in-domain test. Compared with traditional methodologies including W2NER, the State-Of-The-Art model in several ChineseNER tasks, our model significantly outperforms others on the in-domain test set in terms of the Dialogue Act F1 Score. This underscores the exceptional slot-filling accuracy of our model within domain-specific data.

4.2.2 Results on Out-Of-Domain Test

Table 3 showcases the results for the out-of-domain test set. The findings affirm that the average joint accuracy of TransferTOD-7B reached 75.09%, with a Slot F1 Score of 96.20%, surpassing other large-scale models, including the most advanced GPT-4, which only achieved a joint accuracy of 41.68%. In terms of query selection, GPT-4 leads the performance compared to other open-source models. TransferTOD's performance in this aspect scored 75, trailing just behind GPT-4. However, TransferTOD surpassed other models in terms of the fluency of queries. Besides, we have conducted a further experiment to compare our TransferTOD-7B to both open-source and close-source model with In-Context Learning 5-shot setting, reducing the probability of poor score caused by wrong format, the results are presented in Table 15, showing our TransferTOD's superior performance.

The experimental results validate that our TransferTOD model possesses robust generalization capabilities, achieving nearly 80% accuracy in specific downstream tasks. With appropriate secondary fine-tuning, the overall score can be further enhanced.

4.3 Secondary Fine-Tuning Study

4.3.1 Secondary Fine-Tuning

In this section, we primarily discuss our experiments on performing secondary fine-tuning on TransferTOD-7B. The objective was to simulate enhancing our model's slot filling and question-asking capabilities in external scenarios using a small subset of downstream scenario data. We fine-tuned GPT-3.5-Turbo as our baseline and conducted fine-tuning with 50, 100, and 200 pieces of data across three out-of-domain scenarios, respectively. The remaining data served as the test set for this experiment.

In the third scenario (Courier), we undertook multiple experiments employing various fine-tuning strategies, such as adding BELLE (Ji et al., 2023) dataset, incorporating in-domain data, and upsampling out-of-domain scenario data. This research aimed to identify methods that could further enhance the TransferTOD-7B model's slot filling capabilities.

4.3.2 Result

Table 4 shows the results of fine-tuning GPT3.5 and TransferTOD-7B in scenarios. The secondary fine-tuning can improve the model's out-of-domain capability. After fine-tuning, TransferTOD-7B still outperform GPT-3.5 (especially SlotF1) in most cases.

4.4 Ablation Studies

Based on the TransferTOD-v1, v2, v3, and v4 mentioned in 3.1, we trained models TransferTOD-7B-v1 to v4 individually. To ascertain the efficacy and trustworthiness of our data construction methodologies, we rigorously assessed their performance in terms of robustness, diversity and fluency. The method we employed, which combines GPT-based assessment with expert review, is a widely adopted

Model		Scenario	JointAcc(%)	SlotF1(%)	AVG.JointAcc(%)	AVG.SlotF1(%)	Ask_Acc	Ask_Flu
Open-Source Model	**Baichuan2-7B-Chat**	Water-Delivery	15.26	41.83	19.44	44.93	27.50	25.50
		Sanitation	24.29	46.17				
		Courier	18.77	46.79				
	BlueLM-7B-Chat	Water-Delivery	0.80	3.17	0.27	1.06	3.50	0.17
		Sanitation	0.00	0.02				
		Courier	0.00	0.00				
	Chatglm3-6B	Water-Delivery	4.47	23.03	4.11	21.14	25.67	52.67
		Sanitation	4.48	23.99				
		Courier	3.38	16.41				
	Qwen-7B-Chat	Water-Delivery	17.01	38.13	17.14	38.47	28.67	30.67
		Sanitation	16.57	33.45				
		Courier	17.85	43.83				
	Yi-6B-Chat	Water-Delivery	1.04	5.87	1.22	4.59	22.33	52.83
		Sanitation	0.76	2.92				
		Courier	1.85	4.98				
Close-Source Model	**GPT-3.5-Turbo**	Water-Delivery	41.69	74.64	35.71	69.44	72.17	77.67
		Sanitation	31.43	65.44				
		Courier	34.00	68.24				
	GPT-4-1106-Preview	Water-Delivery	42.01	74.21	41.68	70.91	**90.00**	72.33
		Sanitation	40.19	68.32				
		Courier	42.85	70.18				
TransferTOD-7B		Water-Delivery	**73.16**	**96.61**	**75.09**	**96.20**	75.00	**84.00**
		Sanitation	**84.09**	**97.43**				
		Courier	**68.00**	**94.57**				

Table 3: Result of out-of-domain: The Joint Accuracy and Slot F1 Score of each model, showing the accuracy of predicting the right dialogue state and slot-value pairs respectively.

Scenario	Model	Num.ScenarioData	Num.OTD		JointAcc(%)	SlotF1(%)
			TransferTOD	Belle		
Water-Delivery	**GPT-3.5-Turbo**	0	/	/	41.69	74.64
		50	/	/	71.49	93.53
	TransferTOD-7B	0	0	0	73.16	96.61
		50	0	0	**73.48**	**96.64**
Sanitation	**GPT-3.5-Turbo**	0	/	/	31.43	65.44
		100	/	/	78.48	95.78
	TransferTOD-7B	0	0	0	84.09	97.43
		100	0	0	**84.95**	**97.54**
Courier	**GPT-3.5-Turbo**	0	/	/	34.00	68.24
		200	/	/	**78.54**	91.01
	TransferTOD-7B	0	0	0	68.00	94.57
		200	0	0	69.08	94.83
		200×4	8000	0	69.62	95.13
		200×4	8000	8000	68.38	94.81
		200×8	8000	0	70.15	**95.19**

Table 4: Result of Secondary Fine-Tune: The Joint Accuracy and Slot F1 Score of each model, showing the accuracy of predicting the right dialogue state and slot-value pairs respectively. "OTD" stands for Original Train Data which is used in fine-tuning the TransferTOD-7B. "200×4" in Num.ScenarioData represents that we took 200 ScenarioData and repeated it four times.

Model	JointAcc(%)	SlotF1(%)
TransferTOD-7B-v1	11.91	80.53
TransferTOD-7B-v2	55.50	90.24

Table 5: Result of Noise Injection: The Joint Accuracy and Slot F1 Score of each model, showing the accuracy of predicting the right dialogue state and slot-value pairs respectively.

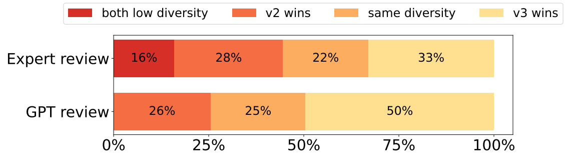

Figure 3: Results of comparative evaluation of TransferTOD-7B-v2 and TransferTOD-7B-v3.

Review Type	0-1 points(%)	2-3 points(%)	Ask_Flu
v3's GPT review	4.50	95.50	95.33
v4's GPT review	2.00	98.00	97.83
v3's Expert review	21.50	78.50	70.50
v4's Expert review	17.50	82.50	75.00

Table 6: Comparison of GPT and Expert Reviews for TransferTOD-7B-v3 and TransferTOD-7B-v4's inquiring fluency. The table shows the proportion of high and low scoring questions in GPT and expert ratings, as well as the corresponding total score.

approach for evaluating the language fluency of models (Chang et al., 2024; Zhang et al., 2024). For details on the GPT assessment instructions and the expert review process, please refer to the appendix B.2, D.2.

Noise Injection To strengthen the model's resilience to noise, we augmented the standard dataset with a controlled amount of noisy data and trained the TransferTOD-7B-v2 model on it. As shown in Table 5, the improvement in joint accuracy substantiates the hypothesis that incorporating noisy data indeed strengthens the model's resistance to noise.

Language Augmentation To enhance the diversity of model interrogation techniques, we expanded our dataset by leveraging GPT, followed by a comprehensive assessment of the diversity in the questions generated by the newly developed models. Evaluators were provided with four assessment options: model A exhibits superior diversity, model B exhibits superior diversity, both models demonstrate comparable diversity, or neither model exhibits satisfactory diversity. The assessment results collected are shown in the Figure 3. Both the outcomes of expert review and GPT review affirm that v3 model surpasses v2 model in linguistic diversity.

Fluency Enhancement To enhance the fluency of the model's inquiries, we manually revised the dataset and employed a hierarchical scoring system to evaluate the models' query smoothness. The findings, as delineated in Table 6, underwent normalization to a 100-point scale, unequivocally demonstrate an improvement in the model's questioning fluency. The calculation of fluency score is given by equation 5.The experimental results demonstrate that the v4 model outperforms v3 model in both the high score rate on the GPT-based review and the expert review, as well as in terms of fluency score. Thus, our method effectively models query fluency.

5 Conclusion

Empirical evidence substantiates that our TransferTOD dataset possesses substantial noise resilience and superior linguistic performance. Utilizing this dataset for supervised fine-tuning, the resultant model, designated TransferTOD-7B, attains a joint accuracy of 75.09% in out-of-domain evaluations, accompanied by a Slot F1 of 96.20%. When it comes to question-asking ability, the accuracy of TransferTOD-7B is only slightly inferior to GPT-4, whereas its fluency in generating questions surpasses all other models we tested.

Furthermore, our findings suggest that appropriate secondary fine-tuning of the TransferTOD-7B model can further enhance its generalization capabilities. By employing a small portion of the out-of-domain test set for secondary fine-tuning, the resulting model surpasses the performance of GPT-3.5-Turbo, which was fine-tuned with an equivalent amount of data.

In summary, we have proposed a highly versatile data construction process that enhances the quality of task-oriented dialogue data for information gathering tasks. Furthermore, the models fine-tuned with this data exhibit strong generalization capabilities, performing well in out-of-domain scenarios.

Limitations

Our research presents a comprehensive set of experiments, yet it is not without limitations. One significant constraint stems from our dataset being primarily in Chinese, which precluded the testing of other major English-language open-source models due to their suboptimal performance on tasks in Chinese. Additionally, our assessment of question-asking accuracy employed manual evaluation methods, potentially introducing a degree of subjectivity despite our efforts to minimize such bias.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Baichuan. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. ArXiv:2005.14165 [cs].

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2020. Multi-WOZ – A Large-Scale Multi-Domain Wizard-of-Oz Dataset for Task-Oriented Dialogue Modelling. ArXiv:1810.00278 [cs].

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. 2024. A survey on evaluation of large language models. *ACM Trans. Intell. Syst. Technol.*, 15(3).

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. 2021. Evaluating Large Language Models Trained on Code. ArXiv:2107.03374 [cs].

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. PaLM: Scaling Language Modeling with Pathways. ArXiv:2204.02311 [cs].

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, and Ziqing Yang. 2021. Pre-training with whole word masking for chinese bert.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2021. All NLP tasks are generation tasks: A general pretraining framework. *CoRR*, abs/2103.10360.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for*

Computational Linguistics (Volume 1: Long Papers), pages 320–335.

Yujie Feng, Zexin Lu, Bo Liu, Liming Zhan, and Xiao-Ming Wu. 2023. Towards LLM-driven Dialogue State Tracking. ArXiv:2310.14970 [cs].

Gabor Fuisz, Ivan Vulić, Samuel Gibbons, Inigo Casanueva, and Paweł Budzianowski. 2022. Improved and Efficient Conversational Slot Labeling through Question Answering. ArXiv:2204.02123 [cs].

Dilek Hakkani-Tür, Gokhan Tur, Asli Celikyilmaz, Yun-Nung Vivian Chen, Jianfeng Gao, Li Deng, and Ye-Yi Wang. 2016. Multi-Domain Joint Semantic Frame Parsing using Bi-directional RNN-LSTM.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS Spoken Language Systems Pilot Corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27,1990*.

Matthew Henderson, Iñigo Casanueva, Nikola Mrkšić, Pei-Hao Su, Tsung-Hsien Wen, and Ivan Vulić. 2020. ConveRT: Efficient and Accurate Conversational Representations from Transformers. ArXiv:1911.03688 [cs].

Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014. The Second Dialog State Tracking Challenge. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 263–272, Philadelphia, PA, U.S.A. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models.

Yunjie Ji, Yong Deng, Yan Gong, Yiping Peng, Qiang Niu, Lei Zhang, Baochang Ma, and Xiangang Li. 2023. Exploring the impact of instruction data scaling on large language models: An empirical study on real-world use cases. *arXiv preprint arXiv:2303.14742*.

Jingye Li, Hao Fei, Jiang Liu, Shengqiong Wu, Meishan Zhang, Chong Teng, Donghong Ji, and Fei Li. 2022. Unified named entity recognition as word-word relation classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 10965–10973.

Xiujun Li, Yun-Nung Chen, Lihong Li, Jianfeng Gao, and Asli Celikyilmaz. 2018. End-to-End Task-Completion Neural Dialogue Systems. ArXiv:1703.01008 [cs].

Zhaojiang Lin, Andrea Madotto, Genta Indra Winata, and Pascale Fung. 2020. MinTL: Minimalist Transfer Learning for Task-Oriented Dialogue Systems. ArXiv:2009.12005 [cs].

Wei Liu, Xiyan Fu, Yue Zhang, and Wenming Xiao. 2021. Lexicon enhanced Chinese sequence labeling using BERT adapter. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5847–5858, Online. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. ArXiv:1907.11692 [cs].

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mo Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela

Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, C. J. Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. 2023. GPT-4 Technical Report. ArXiv:2303.08774 [cs].

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. Generative Agents: Interactive Simulacra of Human Behavior. ArXiv:2304.03442 [cs].

Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayandeh, Lars Liden, and Jianfeng Gao. 2021. Soloist: Building Task Bots at Scale with Transfer Learning and Machine Teaching. *Transactions of the Association for Computational Linguistics*, 9:807–824.

Baolin Peng, Xiujun Li, Lihong Li, Jianfeng Gao, Asli Celikyilmaz, Sungjin Lee, and Kam-Fai Wong. 2017. Composite Task-Completion Dialogue Policy Learning via Hierarchical Deep Reinforcement Learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2231–2240, Copenhagen, Denmark. Association for Computational Linguistics.

Sophie Rosset, Olivier Galibert, and Lori Lamel. 2011. Spoken Question Answering. In *Spoken Language Understanding*, pages 147–170. John Wiley & Sons, Ltd.

Ma Ruotian, Peng Minlong, Zhang Qi, Wei Zhongyu, and Huang Xuanjing. 2020. Simplify the usage of lexicon in chinese ner. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5951–5960.

Haoxiang Su, Hongyan Xie, Hao Huang, Shuangyong Song, Ruiyu Fang, Xiaomeng Huang, and Sijie Feng. 2023. Scalable-DSC: A Structural Template Prompt Approach to Scalable Dialogue State Correction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7902–7914, Singapore. Association for Computational Linguistics.

Xianghui Sun, Yunjie Ji, Baochang Ma, and Xiangang Li. 2023. A comparative study between full-parameter and lora-based fine-tuning on chinese instruction data for instruction following large language model. *ArXiv*, abs/2304.08109.

BlueLM Team. 2023. Bluelm: An open multilingual 7b language model. `https://github.com/vivo-ai-lab/BlueLM`.

Xin Tian, Yingzhan Lin, Mengfei Song, Siqi Bao, Fan Wang, Huang He, Shuqi Sun, and Hua Wu. 2022. Q-tod: A query-driven task-oriented dialogue system.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Hongru Wang, Min Li, Zimo Zhou, Gabriel Pui Cheong Fung, and Kam-Fai Wong. 2021. KddRES: A Multi-level Knowledge-driven Dialogue Dataset for Restaurant Towards Customized Dialogue System. ArXiv:2011.08772 [cs].

Hongru Wang, Lingzhi Wang, Yiming Du, Liang Chen, Jingyan Zhou, Yufei Wang, and Kam-Fai Wong. 2023. A Survey of the Evolution of Language Model-Based Dialogue Systems. ArXiv:2311.16789 [cs].

Tsung-Hsien Wen, David Vandyke, Nikola Mrksic, Milica Gasic, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A Network-based End-to-End Trainable Task-oriented Dialogue System. ArXiv:1604.04562 [cs, stat].

Chien-Sheng Wu, Steven Hoi, Richard Socher, and Caiming Xiong. 2020. TOD-BERT: Pre-trained Natural Language Understanding for Task-Oriented Dialogue. ArXiv:2004.06871 [cs].

Aohan Zeng, Mingdao Liu, Rui Lu, Bowen Wang, Xiao Liu, Yuxiao Dong, and Jie Tang. 2023. Agenttuning: Enabling generalized agent abilities for llms. *ArXiv*, abs/2310.12823.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

Xiaoying Zhang, Baolin Peng, Kun Li, Jingyan Zhou, and Helen Meng. 2023. SGP-TOD: Building Task Bots Effortlessly via Schema-Guided LLM Prompting. ArXiv:2305.09067 [cs].

Yue Zhang, Ming Zhang, Haipeng Yuan, Shichun Liu, Yongyao Shi, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. Llmeval: A preliminary study on how to evaluate large language models. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 19615–19622. AAAI Press.

Kangyan Zhou, Shrimai Prabhumoye, and Alan W. Black. 2018. A Dataset for Document Grounded Conversations. ArXiv:1809.07358 [cs].

Qi Zhu, Kaili Huang, Zheng Zhang, Xiaoyan Zhu, and Minlie Huang. 2020. CrossWOZ: A Large-Scale Chinese Cross-Domain Task-Oriented Dialogue Dataset. ArXiv:2002.11893 [cs].

Jianyun Zou, Min Yang, Lichao Zhang, Yechen Xu, Qifan Pan, Fengqing Jiang, Ran Qin, Shushu Wang, Yifan He, Songfang Huang, and Zhou Zhao. 2021. A Chinese Multi-type Complex Questions Answering Dataset over Wikidata. ArXiv:2111.06086 [cs].

A Experimental Details

A.1 Baselines

For the in-domain test, we select 4 models as baseline:

BertNLU (Zhu et al., 2020) is a BERT-based NLU model, initialized with Chinese pre-trained BERT and fine-tuned on tagged training data. For the input word embeddings, utilize MLP to generate BIO-tagged outputs.

SoftLexicon(LSTM) (Ruotian et al., 2020) is an effective method for incorporating the word lexicon into the character by categorizing the matched words, condensing the word sets and combining them with character representation.

LEBERT+CRF (Liu et al., 2021) Lexicon Enhanced BERT for Chinese sequence labeling, utilizing a Lexicon adapter layer to integrate external lexicon knowledge into BERT layers.

W2NER (Li et al., 2022) is a modeling method of neighboring relations between entity words with Next-Neighboring-Word and Tail-Head-Word-* relations.

For the out-of-domain test, we select 6 Large Language Models as the baseline:

Baichuan2 (Baichuan, 2023) is an open-sourced large language model trained on 2.6 trillion tokens, achieving top performance in various Chinese and multilingual benchmarks. We utilized Baichuan2-7B-chat for our experiments.

ChatGLM3 (Du et al., 2022; Zeng et al., 2022) is Jointly developed by Zhipu AI and Tsinghua University, is the strongest in its class for datasets across multiple disciplines, supporting complex tasks like function calls and code interpretation. We utilized ChatGLM3-6B for our experiments.

Qwen (Bai et al., 2023) Trained on 3 trillion tokens across multiple languages, Qwen models show competitive performance, excelling in tasks like chatting, text generation, and information extraction. We utilized Qwen-7B-chat for our experiments.

Yi [4] A powerful bilingual model, demonstrating significant potential in language cognition and reasoning, ranking highly on the SuperCLUE leaderboard and surpassing other large models in Chinese language proficiency. We utilized Yi-6B-chat for our experiments.

BlueLM (Team, 2023) is a large-scale model from vivo AI Global Research Institute, trained on a 2.6 trillion token corpus, showing leading results in Chinese benchmarks, indicating strong competitiveness. We utilized BlueLM-7B-chat for our experiments.

---

[4] `https://github.com/01-ai/Yi`

GPT-3.5-Turbo [5] stands out as the most potent and cost-efficient model within the GPT-3.5 series. Tailored for conversations, it excels in comprehending and generating natural language.

GPT-4 (OpenAI et al., 2023) is an advanced language model with enhanced understanding and generation capabilities. Trained on diverse internet text, it excels in various tasks, including text generation, translation, and problem-solving. We utilized GPT-4-1106-preview for our experiments.

A.2 Implementation Details

Settings When training TransferTOD-7B, we use Baichuan-7B-base as base model, formatting the data to adapt to the Baichuan training format. Training cost about 8 hours on 8 A800-80GB GPUs and some hyper-parameters of our training are shown in Table 7, each version of our TransferTOD-7B adopted the same hyper-parameters when training.

HyperParameter	Value
num_train_epochs	4
per_device_train_batch_size	1
gradient_accumulation_steps	1
learning_rate	9.65e-6
lr_scheduler_type	cosine
adam_beta1	0.9
adam_beta2	0.98
adam_epsilon	1e-8

Table 7: Hyper-Parameters adopted when training TransferTOD-7B

In-Domain test When training in-domain models with dataset TransferTOD-v4, we tokenize the user utterance with Chinese pre-trained BERT (Cui et al., 2021) and annotate it with sequence labels using BIO tagging scheme.

Out-Of-Domain test For the first part, evaluating the model's capability of slot filling. When inferencing with the LLMs in out-of-domain test, we meticulously designed a system prompt, describing the task and desired output format in detail, to get the best result from each LLM, while some chat models may still perform fairly bad for the slots in their output don't match JSON format. The system prompt used has been translated to English and showed in Table 8.

For the second part, evaluating the semantic accuracy of model-generated questions, we use a manual evaluation approach. Reviewers rate the model's questions on fluency and question accuracy across a scale of 0 to 3 points. A score of 3 for fluency means the model's questioning is semantically coherent, expresses fluently, and fully embodies natural language characteristics, while a score of 0 indicates stiff, awkward language use, not resembling natural language traits. For question accuracy, a full score of 3 implies the model's questions are unambiguous, target slots that are currently empty, and the number of questions matches the given $extract\_slot$ value. In contrast, a score of 0 signifies the model's questions are ambiguous, target slots that have already been filled, or the number of questions doesn't align with the specified $extract\_slot$ value.

A.3 Evaluation Metrics

Joint Accuracy measures the accuracy of dialogue states, considering a state correctly predicted only if all values of given slots are exactly matched.

Given the formula for Joint Accuracy is defined as:

$$JA = \frac{N_{cds}}{T_{ds}} \tag{1}$$

where JA denotes **Joint Accuracy**,N_{cds} stands for the **Number of dialog states correctly predicted**,and T_{ds} represents the **Total number of dialog states**.

Slot F1 calculates the F1 score of (slot, value) pairs, deeming a tuple correctly predicted if the slot's value is exactly matched.
Given the formula for Slot F1 is defined as:

$$\textbf{SlotF1} = \frac{1}{N_{Slots}} \sum_{i=1}^{N_{Slots}} \textbf{F1 Score}_i \tag{2}$$

where N_{Slots} represents the **total number of (slot, value) pairs**.

Dialogue Act F1 calculates the F1 score of (intent, slot, value) dialogue acts, where intent are always "inform", deeming a dialogue act correctly predicted if the slot and value extracted from user utterance is exactly matched. Given the formula

---

[5] https://platform.openai.com/docs/models/gpt-3-5

System

You are an AI responsible for information extraction, and the scenario for information extraction is "<domain>". Based on your conversation with the user, please fill in the slots and continuously ask questions for the slots that are empty, with the number of slots to be asked in each question being <extract_slot>. If the content of the user's answer includes information that does not belong to the slots you asked about in the previous round of conversation, do not fill in the slots with the incorrect parts of the user's answer. Instead, re-ask questions about the incorrect slots in the user's answer.

The format of our input is as follows: Slots: {"Slot_1": "Value_1", "Slot_2": "Value_2", ..., "Slot_n": "Value_n"}
The previous round of conversation: {"assistant": "...", "human": "..."}
If there are still null slots after filling in, your response should follow this format: {"Slot_1": "Value_1", "Slot_2": "Value_2", ..., "Slot_n": "Value_n"}<Questions to ask>
If there are no null slots after filling in, your response should follow this format: {"Slot_1": "Value_1", "Slot_2": "Value_2", ..., "Slot_n": "Value_n"} I have obtained all the information, and here is the content: {"Slot_1": "Value_1", "Slot_2": "Value_2", ..., "Slot_n": "Value_n"}

Table 8: The system prompt used prompting LLMs to execute out-of-domain test, where <domain> represents the domain of the test and <extract_slot> represents the number of slots should be extracted in one turn.

for Dialogue Act F1 is defined as:

$$\textbf{Dialogue Act F1} = \frac{\sum_{i=1}^{N_{DialogueActs}} \textbf{F1 Score}_i}{N_{DialogueActs}} \tag{3}$$

where $N_{DialogueActs}$ represents the **total number of (intent, slot, value) dialogue acts**.

Joint Accuracy is often regarded as a more challenging and significant metric in most literature.

Ask Accuracy measures the model's ability to correctly select the corresponding number of slots from empty slots or to correctly point out errors in user answers and ask questions that correspond to the correct slots and will not cause misunderstandings.

$$\textbf{Ask Accuracy} = \frac{\sum_{i=0}^{3} i \times A_i}{N \times 3} \times 100 \tag{4}$$

where A_i represents the **number of the dialogues that got score i** and N represents the **total number of the dialogues**.

Ask Fluency measures the fluency of the model's questions and the degree to which they are consistent with natural language features.

$$\textbf{Ask Fluency} = \frac{\sum_{i=0}^{3} i \times F_i}{N \times 3} \times 100 \tag{5}$$

where F_i represents the **number of the dialogues that got score i** and N represents the **total number of the dialogues**.

B Prompts

B.1 Prompt GPT-3.5 to Polish the Data

The prompt showed in Table 9 and Table 10 are used when using GPT-3.5 to polish the text, rewriting questions and answers respectively, in our dataset.

User

You are a {domain} company front desk customer service. The following content is the question you want to ask the user. Please change the wording to ask the question. You do not need to output other content, you only need to complete the rewriting.

Original question: {question}
Here's a rephrased version of your question:

Table 9: The prompt for rewriting the question

User

You are a user, the following is the original answer, the specific content name can not be changed, such as the level, service name, etc., please answer in a different expression. You do not need to output other content, just complete the rewrite.

Original answer: {answer}
Here is your answer with a different formulation:

Table 10: The prompt for rewriting the question

B.2 Prompt GPT-4 to Evaluate the Results

The prompts in Table 11 and Table 12 are used when using GPT-4 to score the ask accuracy and ask fluency of the inference results. Because the indicators considered when scoring the ask accuracy of the inference results of clean data and noise data are different, two sets of prompts are used.

The prompt in Table 13 is used when using GPT-4 to conduct comparative evaluation of diversity in ablation experiments, while the prompt in Table 14 is used when scoring the fluency of model questions in ablation experiments with GPT-4.

C Data Examples

Examples of our supervised-finetuning data are showed in Figure 4 and Figure 5, also we provide examples of data with noise in Figure 6 and Figure 7 as well as raw TransferTOD data in Figure 8 and Figure 9.

D Human Experts

D.1 Experts in Constructing Datasets

During the dataset construction phase, we relied on 5 students from our institute to participate in this work as human experts. These students possessed basic computer knowledge and coding skills, which enabled them to perform the task effectively.

Their primary responsibility is to generate dialogue data for test sets. We assign tasks based on different scenarios, ensuring they are familiar with the entire dataset construction process and principles. They work professionally, providing human support for the dataset creation and ensuring smooth project execution. Additionally, we fairly compensate their efforts to show respect and recognition for their contributions.

Another task for human experts involves refining non-fluent content. Given the potential for incoherence and unnaturalness in rule-based generation in 3.1.2, characterized by the lack of appropriate connective words and inconsistent tone, we prioritize addressing this issue.Thus, human experts are employed to revise dialogue content, such as transforming "What's your name? Please tell me your phone number." into a more coherent and natural structure like "Please provide your name and phone number."

Compared to rule-based mass generation, expert-crafted data exhibits significant advantages. The work of domain experts enhances the linguistic fluency, naturalness, and brevity of the generated dialogues. This high-quality, manually constructed data boasts greater authenticity and representativeness, more effectively emulating real-world conversation scenarios. Consequently, it serves as a more reliable foundation for subsequent fine-tuning tasks.

D.2 Experts in Ablation Experiment

During the ablation experiment phase, we invited 12 students from our institution to conduct comparative evaluations of the results. Each student was assigned to complete the full assessment tasks for one or more large models. This entailed each student conducting a comprehensive evaluation of the designated model to ensure a thorough understanding of its performance.

Specifically, we selected 200 data points from the inference results of TransferToD-7B-v2 and TransferToD-7B-v3, and conducted 200 random samples. 5 data points were sampled each time, resulting in a total of 40 evaluations for each model's inference results. This random sampling method contributed to ensuring the objectivity and reliability of the assessment, minimizing potential biases.

Subsequently, the evaluators rated the sampled data based on the questioning style, diversity, and fluency. They provided an overall score for each set of data by considering factors such as the model's questioning approach, sentence completeness, clarity of questioning, diversity, and fluency. These scores provided quantitative data on the model's performance in various aspects, facilitating a more comprehensive assessment and comparison of the models' strengths and weaknesses.

```
{
    "conversations": [
        {
            "from": "human",
            "value": "你是一个负责信息抽取的机器人，需要信息抽取的场景是“租车”。
请你根据与用户的对话填充槽位，并不断对槽位为空的部分进行提问，每一次提问的槽
位数量为 1。如果用户所回答的内容中，有不属于你上轮对话中提问的槽位，那么请不
要将用户回答内容中出错的部分填入槽位，而是对用户回答中出错的槽位进行重新询问。
\n 槽位：{\"平台\": null, \"取车时间\": null, \"还车时间\": null, \"车型\": null, \"预算\": null,
\"车辆价格\": null, \"里程\": null, \"姓名\": null, \"地址\": null, \"身份证号\": null}\n 上轮对
话：{\"assistant\"：\"请告诉我您的租车预算。\",\"human\"：\"我的租车预算是 465.1 元。
\"}"
        },
        {
           "from": "assistant",
            "value": "{\"平台\": null, \"取车时间\": null, \"还车时间\": null, \"车型\": null,
\"预算\": 465.1, \"车辆价格\": null, \"里程\": null, \"姓名\": null, \"地址\": null, \"身份证号\":
null} 您选择的车辆价格是？"
        }
    ]
}
```

Figure 4: An example of the training data for supervised-finetuning TransferTOD-7B

```
{
    "conversations": [
        {
            "from": "human",
              "value": "You are a robot responsible for information extraction. The scenario
that requires information extraction is "car rental". Please fill in the slots based on the
conversation with the user, and continue to ask questions for the empty slots. The number of
slots for each question is 1. If the content of the user's answer does not belong to the slot you
asked in the previous round of dialogue, then please do not fill in the wrong part of the user's
answer into the slot, but fill in the wrong slot in the user's answer. Requery.\n Slot:
{\"Platform\": null, \"Pick-up time\": null, \"Return time\": null, \"Car model\": null, \"Budget\
": null, \"Vehicle Price\": null, \"Mileage\": null, \"Name\": null, \"Address\": null, \"ID
Number\": null}\nUp Turn dialogue: {\"assistant\": \"Please tell me your car rental budget.
\",\"human\": \"My car rental budget is 465.1 yuan. \"}"
        },
        {
           "from": "assistant",
            "value": "{\"Platform\": null, \"Pick-up time\": null, \"Return time\": null, \"Car
model\": null, \"Budget\": 465.1, \"Vehicle price\ ": null, \"mileage\": null, \"name\": null,
\"address\": null, \"ID number\": null} What is the price of the vehicle you selected?"
        }
    ]
}
```

Figure 5: An example of the training data for supervised-finetuning TransferTOD-7B (in English)

```
{
        "conversations": [
            {
                "from": "human",
                "value": "你是一个负责信息抽取的机器人，需要信息抽取的场景是“酒店”。
请你根据与用户的对话填充槽位，并不断对槽位为空的部分进行提问，每一次提问的槽
位数量为2。如果用户所回答的内容中，有不属于你上轮对话中提问的槽位，那么请不
要将用户回答内容中出错的部分填入槽位，而是对用户回答中出错的槽位进行重新询问。
\n槽位：{\"酒店类型\": \"舒适型\", \"酒店服务\": \"酒廊\", \"酒店星级\": null, \"入住日期\":
\"下周三\", \"入住时间\": null, \"入住天数\": null, \"是否需要吸烟房\": null, \"姓名\": \"虞
新荣\", \"身份证\": \"356053197405022073\", \"手机号\": \"1503676461\"}\n上轮对话：
{\"assistant\"：\"您当天预计何时办理入住呢？您想要住几星级酒店呢？\",\"human\"：\"
大概在傍晚6点左右。不用订吸烟房\"}"
            },
            {
                "from": "assistant",
                 "value": "{\"酒店类型\": \"舒适型\", \"酒店服务\": \"酒廊\", \"酒店星级\":
null, \"入住日期\": \"下周三\", \"入住时间\": \"傍晚6点\", \"入住天数\": null, \"是否需要吸
烟房\": null, \"姓名\": \"虞新荣\", \"身份证\": \"356053197405022073\", \"手机号\":
\"1503676461\"} 对不起，您可能理解错了我的意思。您对于酒店的星级有什么要求吗？
"

            }
        ]
}
```

Figure 6: An example of the training data with noise for supervised-finetuning TransferTOD-7B

```
{
        "conversations": [
            {
                "from": "human",
                 "value": "You are a robot responsible for information extraction, and the scene
that requires information extraction is "hotel". Please fill the slots based on the conversation
with the user, and continue to ask questions in the empty slots. The number of slots for each
question is 2. If there is a slot in the user's answer that does not belong to the slot you asked in
the previous round of dialogue, then please do not fill in the wrong part of the user's answer
into the slot, but re-ask the wrong slot in the user's answer. \nSlot: {\"Hotel Type\":
\"Comfort\", \"Hotel Service\": \"Lounge\", \"Hotel Star\": null, \"Check-in Date\" : \"Next
Wednesday\", \"Check-in Time\": null, \"Number of Days\": null, \"Smoking Room\": null,
\"Name\": \"Yu Xinrong\", \ "ID card\": \"356053197405022073\", \"Phone Number\":
\"1503676461\"}\nLast round of conversation: {\"assistant\": \"When do you plan to check in
that day? Which star hotel do you want to stay in? \",\"human\": \"No need to book a smoking
room\"}."
            },
            {
                "from": "assistant",
                 "value": "{\"Hotel Type\": \"Comfort\", \"Hotel Service\": \"Lounge\", \"Hotel
Star\": null, \"Check-in Date\": \"Next Wednesday \", \"Check-in Time\": \"6pm\", \"Number of
Days\": null, \"Smoking Room\": null, \"Name\": \"Yu Xinrong\", \"ID card\":
\"356053197405022073\", \"Phone Number\": \"1503676461\"} Sorry, you may have
misunderstood what I meant. Do you have any requirements for hotel star rating?"
            }
        ]
}
```

Figure 7: An example of the training data with noise for supervised-finetuning TransferTOD-7B (in English)

```
{
      "id": 1,
      "task": "酒店",
      "extract_slot": 1,
      "content": [
            ...
    {
            "Turn": 1,
            "origin_slots": {
                "酒店类型": null,
                "酒店服务": null,
                "酒店星级": null,
                "入住日期": null,
                "入住时间": null,
                "入住天数": null,
                "是否需要吸烟房": null,
                "姓名": null,
                "身份证": null,
                "手机号": null
            },
            "conversations": [
                {
                    "from": "user",
                    "value": "帮我定一个 1 星级的酒店"
                },
                {
                    "from": "assistant",
                    "value": "您的身份证号码是什么？ "
                }
            ],
            "new_slots": {
                "酒店类型": null,
                "酒店服务": null,
                "酒店星级": 1,
                "入住日期": null,
                "入住时间": null,
                "入住天数": null,
                "是否需要吸烟房": null,
                "姓名": null,
                "身份证": null,
                "手机号": null
            }
    }
            ...
```

Figure 8: An example of our TransferTOD dataset

```
{
        "id": 1,
        "task": "Hotel",
        "extract_slot": 1,
        "content": [
                ...
     {
                "Turn": 1,
                "origin_slots": {
                    "Hotel Type": null,
                    "Hotel Service": null,
                    "Hotel Star": null,
                    "Check-in Date": null,
                    "Check-in Time": null,
                    "Number of Days": null,
                    "Smoking Room": null,
                    "Name": null,
                    "ID card": null,
                    "Phone Number": null
                },
                "conversations": [
                    {
                        "from": "user",
                        "value": "Help me book a one-star hotel"
                    },
                    {
                        "from": "assistant",
                        "value": "What is your ID number?"
                    }
                ],
                "new_slots": {
                    "Hotel Type": null,
                    "Hotel Service": null,
                    "Hotel Star": 1,
                    "Check-in Date": null,
                    "Check-in Time": null,
                    "Number of Days": null,
                    "Smoking Room": null,
                    "Name": null,
                    "ID card": null,
                    "Phone Number": null
                }
     }
                ...
```

Figure 9: An example of our TransferTOD dataset (in English)

User

This text describes a conversational scenario for completing an information extraction task. In this scenario, a customer service representative is required to continually extract information from fields that have not yet been filled, asking questions based on a specified number of fields (denoted as {extract_slot}) to be extracted. If the number of remaining unfilled fields is less than {extract_slot}, the representative should pose questions covering all remaining fields. The representative must then extract relevant information from the customer's responses to fill in the corresponding fields. If all the fields have been filled in, the customer service needs to output "I have obtained all the information, the following is the information content:", and then give all the fields that have been filled in in the form of a json string. In instances where a customer's response is either irrelevant, contains common sense errors, or other inaccuracies, the representative must identify these errors and re-ask questions related to the fields where errors occurred in the previous round. The performance of the customer service representative in this scenario is to be evaluated based on two criteria: fluency of questioning and accuracy of questioning.

Fluency of Language:

- 0 points represent that the representative's questioning style is rigid and awkward, completely deviating from the characteristics of natural language.
- 1 point represent that the representative's questioning style is somewhat rigid, yet the language is relatively natural, aligning with certain characteristics of natural language.
- 2 points represent that the representative's questioning style is relatively natural, and the language used is also quite consistent with the characteristics of natural language.
- 3 points represent that the representative's questioning style is very natural, and the language fully complies with the characteristics of natural language.

Accuracy of Questioning:

- 0 points represent that the model's questions are ambiguous, or it fails to correctly select fields from the empty slots for questioning, and the number of questioned fields does not match {extract_slot} (if the number of remaining empty fields is less than {extract_slot} and the number of questions asked does not equal the total of all remaining empty fields while meeting the previous condition, it should also be categorized here).
- 1 point represent that the model's questions might cause ambiguity, but it can correctly select fields from the empty slots for questioning, yet the number of questioned fields does not match {extract_slot} (if the number of remaining empty fields is less than {extract_slot}) and the number of questions asked does not equal the total of all remaining empty fields while meeting the first two conditions, it should also be categorized here).
- 2 points represent that the model's questions are precise, unambiguous, and it can correctly select fields from the empty slots for questioning, but the number of questioned fields does not match {extract_slot}) (if the number of remaining empty fields is less than {extract_slot} and the number of questions asked does not equal the total of all remaining empty fields while meeting the first two conditions, it should also be categorized here).
- 3 points represent that the model's questions are precise, unambiguous, and it can correctly select fields from the empty slots for questioning, and the number of questioned fields matches {extract_slot}) (if the number of remaining empty fields is less than {extract_slot}) and the number of questions asked equals the total of all remaining empty fields while meeting the first two conditions, it should also be categorized here). If all slots are filled and the model does not initiate a question or says "I have obtained all the information," the message content "" should also fall into this category.

The current slot information for the scenario is as follows: {slot}
The customer service's question is as follows: {ques}

You must provide your feedback in the following format:
Reason: reason
Accuracy: score of its accuracy (int)
Fluency: score of its fluency (int)

Table 11: The prompt used for using GPT-4 to score the ask accuracy and ask fluency of the inference results of clean data.

User
The following is a dialog scenario for completing an information extraction task, where customer service representatives continually extract fields from yet-to-be-filled fields to ask questions, and extract the corresponding information from the customer's answers to fill in the respective fields. If all the fields have been filled in(the value corresponding to no field is null), the customer service needs to output "I have obtained all the information, the following is the information content:", and then give all the fields that have been filled in in the form of a json string. If the customer's answer contains errors such as irrelevant responses or commonsense mistakes, the customer service representative needs to point out the corresponding errors. They need and only need to re-ask once about the field that contained errors in the previous round. In the example provided, the user's answer contained errors, where the last round of dialog content from the user is represented as last, "gpt" represents the customer service response, and "human" represents the user's answer. You need to score the performance of the customer service in this scenario in terms of the fluency of the questioning and the accuracy of the questioning: **Fluency of Language:** - 0 points represent that the customer service's error identification and questioning manner are awkward, the language is convoluted, and completely inconsistent with the characteristics of natural language. - 1 point represents that the customer service's error identification and questioning manner are awkward, but the language is relatively natural and conforms to certain characteristics of natural language. - 2 points represent that the customer service's error identification and questioning manner are relatively natural, and the language also conforms more to the characteristics of natural language. - 3 points represent that the customer service's error identification and questioning manner are very natural, and the language fully conforms to the characteristics of natural language. **Accuracy of Questioning:** - 0 points represent that the model's questioning is ambiguous, or the model cannot correctly point out the errors in the user's answer and re-ask. - 1 point represents that the model's questioning might cause ambiguity, or the model can point out the errors in the user's answer but cannot re-ask about the erroneous field (i.e., there is an issue with the field being questioned). - 2 points represent that the model's questioning is accurate, does not cause ambiguity, and the model can point out the errors in the user's answer, but cannot re-ask about the erroneous field (i.e., there is an issue with the field being questioned). - 3 points represent that the model's questioning is accurate, does not cause ambiguity, and the model can point out the errors in the user's answer and re-ask about the corresponding field. The current slot information for the scenario is as follows: {slot} The customer service's question is as follows: {ques} You must provide your feedback in the following format: Reason: reason Accuracy: score of its accuracy (int) Fluency: score of its fluency (int)

Table 12: The prompt used for using GPT-4 to score the ask accuracy and ask fluency of the inference results of noise data.

User

The following is a dialogue scenario for a task of information extraction, where two customer service representatives are inquiring customer information. You are required to compare the diversity in questioning styles and sentences between two groups in order to evaluate their performance.

Your options are as follows:
Option A: Group A's questioning style is noticeably more diverse than Group B's.
Option B: Group B's questioning style is noticeably more diverse than Group A's.
Option C: Both Group A and Group B demonstrate a similar level of diversity in their questioning.
Option D: Both Group A and Group B lack diversity in their questioning.

The inquiries from customer service A are as follows: {selected_a_questions}
The inquiries from customer service B are as follows: {selected_b_questions}

You must provide your feedback in the following format:
Reason: reason
Option: A, B, C or D

Table 13: The prompt used when using GPT-4 to conduct comparative evaluation of diversity in ablation experiments.

User

The following scenario is a customer service question asked by a user to obtain specific information. You need to rate the fluency of the customer service question. Fluency includes factors such as whether the question is a complete sentence, whether it contains pauses of unclear meaning, whether the questioning method is blunt, whether it conforms to the characteristics of natural language, etc., and customer service questions are scored accordingly. If the customer service says "I have obtained all the information, the following is the information content" and is followed by a json string, the item will be rated as a full score.

Fluency:
- 0 points mean that the customer service's questions are not fluent. Multiple questions are divided into many independent questions, or contain pauses with unclear meaning. The questioning method is stiff. Completely inconsistent with the characteristics of natural language
- 1 point means that the customer service questions are not fluent. Multiple questions are divided into multiple short sentences, or contain relatively abrupt pauses. Not consistent with the characteristics of natural language
- 2 points mean that the customer service questions are relatively fluent, and multiple questions are relatively fluently combined into long sentences, which is more in line with the characteristics of natural language.
- 3 points mean that the customer service questions are very fluent, and multiple questions are fluently combined into long sentences, which fully conforms to the characteristics of natural language.

The customer service question content is as follows: {ques}

You must give your feedback in the following format:
Reason: reason
Fluency: score of its fluency (int)

Table 14: The prompt used when scoring the fluency of model questions in ablation experiments using GPT-4.

Model	Scenario	JointAcc(%)	SlotF1(%)	AVG.JointAcc(%)	AVG.SlotF1(%)
TransferTOD-7B	Water-Delivery	**75.16**	**96.61**	**75.09**	**96.20**
	Sanitation	**84.09**	**97.43**		
	Courier	**68.00**	**94.57**		
Baichuan2-7B-Chat(5-shot)	Water-Delivery	52.40	82.16	53.78	82.42
	Sanitation	71.71	94.92		
	Courier	37.23	70.19		
BlueLM-7B-Chat(5-shot)	Water-Delivery	61.98	93.87	42.81	86.32
	Sanitation	43.90	87.54		
	Courier	22.54	77.57		
Chatglm3-6B(5-shot)	Water-Delivery	22.92	53.35	27.32	64.24
	Sanitation	31.43	67.42		
	Courier	27.62	71.96		
Qwen-7B-Chat(5-shot)	Water-Delivery	69.09	94.04	61.69	91.44
	Sanitation	61.14	91.26		
	Courier	54.85	89.02		
Yi-6B-Chat(5-shot)	Water-Delivery	67.89	94.94	63.09	94.04
	Sanitation	64.00	93.87		
	Courier	57.38	93.32		
GPT-4-1106-Preview(5-shot)	Water-Delivery	65.10	75.98	65.39	76.47
	Sanitation	65.14	75.87		
	Courier	65.92	77.57		

Table 15: Result of out-of-domain with the setting of In-Context Learning: The Joint Accuracy and Slot F1 Score of each model, showing the accuracy of predicting the right dialogue state and slot-value pairs respectively.