Exploring Hallucinations in Task-oriented Dialogue Systems with Narrow Domains

Yan Pan
Technical University of Munich
frankpanyan96@gmail.com

Davide Cadamuro
BMW Group
davide.cadamuro@bmw.de

Georg Groh
Technical University of Munich
grohg@in.tum.de

Abstract

Task-oriented dialogue systems with large language models (LLMs) show powerful language capabilities. These systems aim to solve particular tasks in narrow domains and consist of different modules. These modules, such as the dialogue state tracker or the response generator, are powered by LLMs. However, LLMs like ChatGPT are prone to hallucinations, which are challenging to spot. This is due to the complex nature of the systems and the limited datasets for narrow domains. This phenomenon could have dangerous consequences for the user, which motivates us to study the hallucination problem. Our task-oriented dialogue hallucination study consists of situation analysis, dataset generation, and hallucination detection in different modules within narrow domains. We analyze the hallucination situation for different modules based on the collected hallucination samples from ChatGPT. We obtain high hallucination rates among modules. Due to the shortage of hallucination datasets, we propose a hallucination score to build suitable hallucination samples from existing datasets. Moreover, we present a Task-oriented Hallucination Detector (THD) for the different modules and domains, which benefits from the generated hallucination samples.

1 Introduction

Large language models (LLMs) show their powerful capabilities in task-oriented dialogue systems, which are widely used to help people solve specific tasks, ranging from booking a hotel to finding a restaurant with a given domain knowledge. Recently, researchers utilized LLMs as the backbone for different modules in task-oriented dialogue systems, such as the dialogue state tracker (Hu et al., 2022b) or the response generator (Hudeček and Dusek, 2023). However, recent black-box LLMs, such as ChatGPT (OpenAI, 2022), tend to generate hallucinations, i.e., they are unfaithful to the domain knowledge or to the information provided by the user (Bang et al., 2023). These hallucinations may provide misleading information or even lead to dangerous situations for the end-user (Li et al., 2023). Therefore, it is imperative and valuable to study the hallucination problem in task-oriented dialogue systems.

In comparison to chit-chat chatbots, LLM-based task-oriented dialogue systems require a state representation to query the domain-related knowledge base (Zhang et al., 2020). A typical system consists of a pipeline of different modules, such as a domain detector, a dialogue state tracker, a dialogue policy, and a response generator, shown as gray blocks in Figure 1 (Zhang et al., 2020; Hudeček and Dusek, 2023). The pipeline of different components is more explainable, controllable, and easier to implement than the end-to-end approach, which uses a unified model (Kwan et al., 2023). However, the complex architecture of a task-oriented dialogue system further complicates the hallucination problem since hallucinations can affect each part of this pipeline.

Figure 1 presents one dialogue example from a task-oriented dialogue dataset, namely MWOZ 2.1 (Eric et al., 2020; Budzianowski et al., 2018). In this task-oriented dialogue, the user wants to find a restaurant called Prezzo. To accomplish the goal, the LLM domain detector first detects the current domain of the user's query. Consequently, the instructions of the pipeline are determined by the predicted domain. Then, the LLM state tracker extracts the user's intention and presents it as a slot and value pair (Hu et al., 2022b). Based on the captured slot and value pair, the task-oriented dialogue system searches for a restaurant called Prezzo from the domain-related knowledge database, which contains information on restaurants. With the retrieved restaurant information, the LLM dialogue policy decides which assistant actions to take. Finally, the LLM response generator creates a response based on the correct dialogue actions.

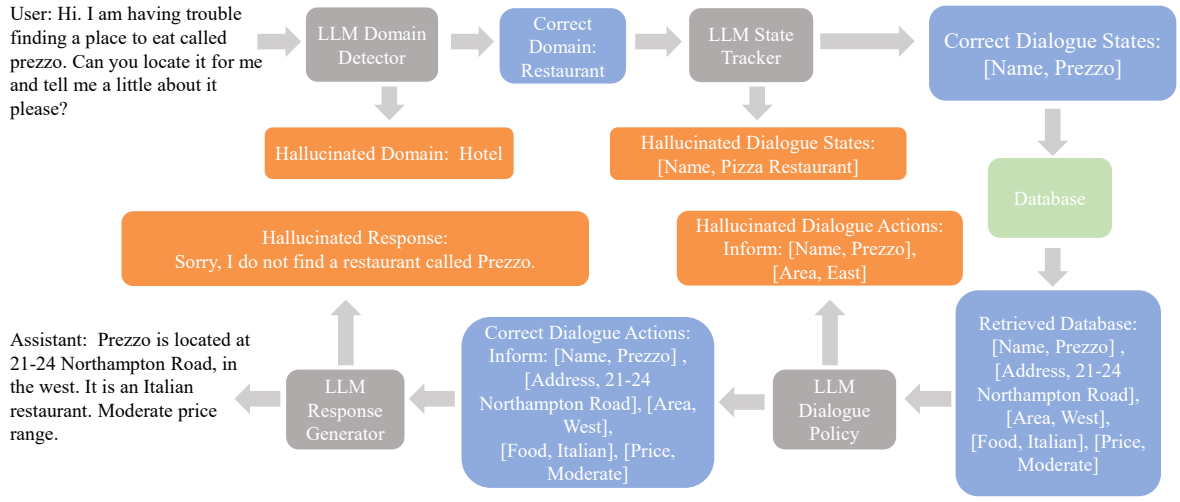

Figure 1: A task-oriented dialogue example from the MWOZ 2.1 dataset (Eric et al., 2020; Budzianowski et al., 2018) with correct outputs (blue boxes) and simulated hallucination outputs (orange boxes) for the domain detector, dialogue state tracker, dialogue policy, and dialogue response generator (gray boxes).

Figure 1 also shows a plausible example of hallucination for each module. A hallucinated domain classification has a detrimental impact on the entire execution, since a task-specific pipeline is selected at this stage to complete the assignment. Hallucinated dialog states result in ineligible restaurants being retrieved from the database. Hallucinated dialog actions from the LLM dialog policy contradict the retrieved restaurant information. Finally, there is a risk that the user will receive a hallucinated response due to the response generation module.

In this paper, we study the hallucination problem for all modules of the black-box-LLM-based task-oriented dialogue systems with narrow domains. Our study framework consists of situation analysis, dataset generation, and hallucination detection.

For situation analysis, we collected naturally generated hallucination samples from ChatGPT to analyze how the different modules are affected by hallucinations. The analysis results show that there are many forms of undesirable hallucinations, with LLM-based modules suffering from hallucination rates of up to 28.8%. Therefore, hallucination detection for task-oriented modules is a critical task.

Current datasets for hallucination detection are limited to just dialogue responses and are not suitable for all modules. Annotating hallucination samples is expensive and time-consuming. However, researchers propose numerous multi-domain task-oriented dialogue datasets. We propose a method with a hallucination score which automatically builds hallucination samples from these existing datasets, to overcome the dataset shortage problem, as shown in Figure 2. For each sample with input materials and a correct output, outputs from other samples can be considered as hallucination output candidates. The hallucination score measures the relatedness between the input materials and output candidate and the similarity between the correct output and output candidate. The output candidate with high relatedness and low similarity is selected as a hallucination output.

Finally, we present our Task-oriented Hallucination Detector (THD) as shown in Figure 3, which is a fine-tuned DistilBERT-based classifier (Sanh, 2019; Wolf et al., 2020) with Low-Rank Adaptation (LoRA) (Hu et al., 2022a). The classifier is fine-tuned with the generated hallucinated samples and existing correct samples. THD learns the forms of hallucination among different modules and domains through fine-tuning. Moreover, LoRA is added to the fine-tuned DistilBERT-based classifier to achieve better performance for different modules and domains. The experimental results on MWOZ 2.1 and M2M (Shah et al., 2018) datasets indicate that our THD outperforms other hallucination detection models.

To the best of our knowledge, this work is the first attempt to explore hallucination generation and detection framework for all black-box-LLM-based modules in task-oriented dialogue systems with narrow domains. Our main contributions are three-fold:

- We conducted hallucination situation analysis based on collected real samples, which show

non-negligible hallucination rates and forms for different modules.

- We propose a method with a hallucination score to automatically build hallucination samples, which is widely applicable in different narrow domains.

- The hallucination detection experimental results on the generated hallucinated MWOZ 2.1 and M2M datasets show that overall our THD can achieve higher accuracy than other evaluated hallucination detection methods for task-oriented dialogue hallucination problems.

2 Related Work

2.1 Hallucination from Large Language Models

Hallucinations could result in the spread of false information and raise serious risks in specific domains (Ji et al., 2023), for example, inaccurate medical information from LLMs (Sharun et al., 2023). These hallucinations are unfaithful or nonsensical texts generated by generative models, which give the natural impression (Ji et al., 2023). For task-oriented dialogue, the generated text is based on the source content, including the instruction, dialogue information, and domain-related knowledge base. The hallucination in task-oriented dialogue emphasizes the inconsistency of generated text from the provided source content (Huang et al., 2023).

2.2 Hallucination Benchmark

To study hallucination from LLMs, researchers have proposed some dialogue-related benchmarks in recent years (Li et al., 2023; Chen et al., 2024; Dziri et al., 2022). However, the annotation for these hallucination benchmarks is very challenging, time-consuming, and expensive. Due to the diverse hallucination instances and ambiguous contents, annotators need high levels of expertise (Chen et al., 2024). Li et al. (2023) utilized labelers with good reading comprehension to annotate generated hallucination response samples. Moreover, these hallucination benchmarks are limited to dialogue responses instead of whole modules of task-oriented dialogue systems (Li et al., 2023; Chen et al., 2024). This paper studies the hallucination problem among all modules and proposes an efficient method for the automatic generation of hallucination samples.

2.3 Hallucination Detection

Recently developed generative LLMs are often released as black-boxes accessed through APIs (OpenAI, 2022; Achiam et al., 2023). These black-box LLMs are used as the backbones for different modules in task-oriented dialogue systems (Hudeček and Dusek, 2023; Bang et al., 2023). Li et al. (2023) utilized GPT3 (Brown, 2020), and ChatGPT to detect hallucinations in open-domain dialogue responses. GPT4 (Achiam et al., 2023) also shows powerful hallucination detection capability in task-oriented dialogue responses (Chen et al., 2024). This paper focuses on hallucination detection from black-box-LLM-based modules in task-oriented dialogue systems.

3 Study Framework

Our study framework focuses on the hallucination problem in all the modules of task-oriented dialogue systems with narrow domains. It consists of three main parts: (1) hallucination situation analysis, (2) hallucination dataset generation, and (3) hallucination detector development.

3.1 Task-oriented Hallucination Analysis

To find the real hallucination incidences in all task-oriented dialogue modules, ChatGPT is employed to generate domain prediction, dialogue states, actions, and responses following Hudeček and Dusek (2023) and Zhang et al. (2020). The task instruction describes the specific requirements and examples for each module and each narrow domain. The input prompt consists of the task instruction and the corresponding input materials as shown in Table 2, like the dialogue context, the dialogue states, the database information, and the dialogue actions. The input prompt is fed into ChatGPT to generate the module output, which is then annotated by human labelers. They detect whether the generated output contains hallucinated content. We collect three labels for each module output. The max-voting label result determines the final hallucination label.

3.2 Task-oriented Hallucination Generation

After the situation analysis, our framework uses an existing dataset to build a task-oriented hallucinated output. As shown in Figure 2, each sample from the existing dataset contains input materials and a corresponding correct output. Inspired by Karpukhin et al. (2020), all other output in the ex-

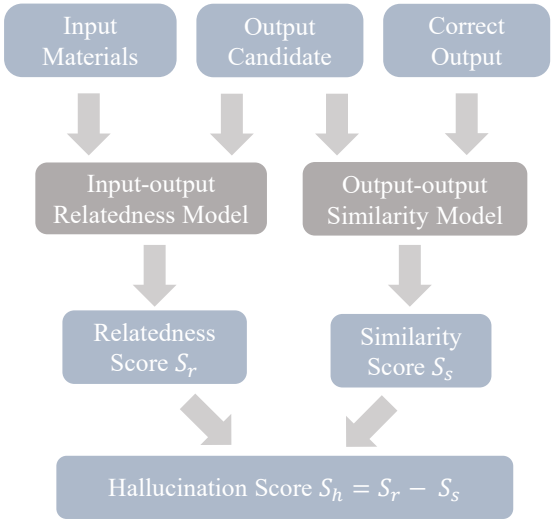

Figure 2: Task-oriented hallucination generation.

isting dataset can be considered as hallucinated output candidates. To ensure the high quality of hallucinated samples, on the one hand, the hallucinated output should be semantically related to the input materials. We build an input-output relatedness model to measure the relatedness score S_r. On the other hand, the hallucinated output should be different from the correct output. The output-output similarity model measures the similarity score S_s between the output candidate and the correct output. Therefore, for each output candidate, we define the corresponding hallucination score $S_h = S_r - S_s$ to measure how suitable the candidate is as a hallucinated output for this sample. After ranking, the most suitable output candidate with the highest hallucination score is selected as the hallucinated output. Based on different splits of existing datasets, the framework builds training, validating, and testing hallucinated samples. Combining correct and hallucinated outputs, we get the dataset for the following hallucination detection task.

The framework uses the sentence transformer model (Reimers and Gurevych, 2019) as an input-output relatedness model and an output-output similarity model. The input-output relatedness model maps the input materials into the representation e_i, and the output candidate into the e_{cr}. The relatedness score S_r is measured by the similarity between e_i and e_{cr}. The output-output similarity model maps the correct output into e_o, and the output candidate into the e_{cs}. The similarity score S_s is measured by the similarity between e_o and e_{cs}. To obtain accurate scores, these models are fine-tuned using positive and negative samples from the existing dataset. For the relatedness model, positive samples consist of input materials and correct outputs. Negative samples contain input materials and randomly sampled outputs. For the similarity model, we use back-translation (Sennrich et al., 2016) to augment the rewritten output, translating the correct output into another language and then back to English. The positive samples then consist of the correct and rearranged outputs. Negative samples consist of correct and randomly sampled outputs.

3.3 Task-oriented Hallucination Detection

We designed a Task-oriented Hallucination Detector (THD) to tackle hallucination detection in different task-oriented dialogue modules. As shown in Figure 3, the output and input materials from each sample are fed into the DistilBERT-based classifier to get the representation $e = DistilBERT([Output, InputMaterials])$. Based on the representation e, the classifier predicts if the output contains hallucination. The DistilBERT-based classifier is fine-tuned with samples of existing correct outputs and generated hallucination outputs from all domains.

After fine-tuning using samples from all domains, we add the LoRA (Hu et al., 2022a) into the fine-tuned classifier for each module and domain. LoRA keeps the DistilBERT-based classifier parameters frozen. The model layer with the form $h = W_0x$ is re-parameterized as $h = W_0x + \frac{\alpha}{r}BAx$. The $W_0 \in R^{d\times k}$, x, and h represent the weight matrix, input, and output, respectively. The $B \in R^{d\times r}$ and $A \in R^{r\times k}$ are the decomposition matrices, which contain trainable parameters. r represents the rank of the decomposition, and α is a constant (Hu et al., 2022a; Poth et al., 2023; Pfeiffer et al., 2020). The model with the LoRA adapter is fine-tuned with corresponding samples from the module and domain. The LoRA is implemented for the detector to analyze the output from the dialogue state tracker, the dialogue policy, and the response generator.

4 Experiments

4.1 Datasets

We conducted our experiments on two multi-domain task-oriented dialogue datasets, MWOZ 2.1 (Eric et al., 2020; Budzianowski et al., 2018) and M2M (Shah et al., 2018). These datasets are

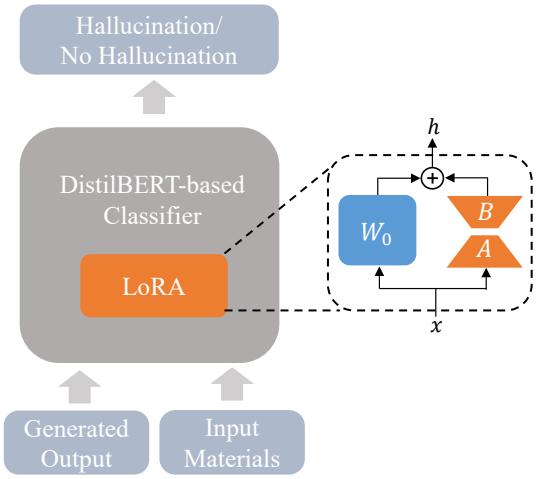

Figure 3: Task-oriented Hallucination Detector (THD) with LoRA (Hu et al., 2022a).

widely used benchmarks for evaluating different dialogue modules. For the MWOZ 2.1 dataset, we selected the following five domains: restaurant, hotel, train, taxi, and attraction. The M2M dataset contains dialogues spanning movie and restaurant domains. However, we skipped the dialog policy module for M2M, because there is no explicit database for this dataset (Shah et al., 2018).

For the hallucination situation analysis, we sampled 500 cleaned samples from MWOZ 2.1. In this dataset, each domain contains 100 cleaned samples. For dataset generation, we sampled 2000/1000/1000 correct samples from the MWOZ 2.1 train/dev/test set. Each correct sample contains input materials and output for all modules. Our framework created a hallucination sample with a hallucinated output for each correct sample. After combining the correct and hallucinated outputs, we obtained 4000/2000/2000 samples for MWOZ 2.1. Following the same procedure, we obtained 1600/800/800 samples for M2M from 800/400/400 correct samples.

4.2 Experimental Details

For our hallucination situation analysis, ChatGPT was used to generate outputs from all modules in task-oriented dialogue systems. For the dataset generation, we utilized the sentence transformer model "All-mpnet-base-v2" (Reimers and Gurevych, 2019; Song et al., 2020) as the backbone for both the input-output relatedness model and the output-output similarity model. ChatGPT was used for back-translation to augment the rewritten outputs. For the hallucination detection, we fine-tuned the DistilBERT model, which is a transformer-based encoder model. Similarly to Li et al. (2023), we report accuracy in determining whether a sample contains hallucinated information, to evaluate hallucination detection models.

	Domain	State	Action	Response
Number	30	85	144	88
Rate	6.0%	17.0%	28.8%	17.6%

Table 1: Hallucination rate statistic of 500 ChatGPT outputs on the MWOZ 2.1 dataset for different modules.

4.3 Existing Hallucination Detection Models

Many recent studies use different LLMs to detect hallucinations (Li et al., 2023; Chen et al., 2024). In this paper, we tested the following models for hallucination detection:

- ChatGPT: A model introduced by OpenAI utilizes reinforcement learning from human feedback (OpenAI, 2022).
- Command R: An LLM optimized for long context tasks shows strong performances on retrieval generation tasks (Cohere, 2024).
- GPT4: The advanced model from OpenAI presents advanced reasoning capability and great performance on many natural language processing tasks (Achiam et al., 2023).

5 Results and Discussion

5.1 Hallucination Situation Analysis

Table 1 presents the statistics of hallucination rates among all modules. For all modules, we manually sample examples of an input, a correct output, a ChatGPT output, and a generated hallucination output. All these examples are described in Table 2 to show the forms of different modules. Both tables show that all modules suffer from hallucination problems.

Table 1 shows that from 6.0% up to 28.8% of ChatGPT outputs are hallucinated, depending on the module that produced the samples. These values indicate that hallucination detection is a critical problem for task-oriented dialogue systems. For a deeper understanding of the hallucinations, we conducted the following analysis for the different modules based on the statistics and real hallucination examples.

Domain Detector	
Input Materials	Context: Customer: help me get a taxi to the cambridge museum of technology please.
Correct Output	Taxi
ChatGPT Output	**Attraction**
Gen. Hallucination	**Attraction**
Dialogue State Tracker	
Input Materials	Context: ... Customer: Yes, I'm looking for a museum in the west. Assistant: How does the cafe jello gallery sound? Or kettles yard? Customer: Could I get the area, phone number, and address of the Kettles yard?
Correct Output	Attraction: [Type, Museum], [Name, Kettles yard], [Area, West]
ChatGPT Output	**Attraction: [Type, Museum], [Area, Kettles yard]**
Gen. Hallucination	**Attraction: [Type, Don't care], [Area, West]**
Dialogue Policy	
Input Materials	Context: Customer: Hi, I'm trying to find a moderate place to eat in the east. Can you help? Assistant: I've found a place called curry prince that fits that. Would you like me to place a reservation for you? Customer: Actually I'm more interested in middle Eastern foods. States: Restaurant: [Food, Middle eastern], [Pricerange, Moderate], [Area, East] Database: No suitable restaurant
Correct Output	Request: [Food, ?]; No-offer: [Food, Middle east]
ChatGPT Output	**Recommend: [Food, Middle eastern]; Inform: [Food, Middle eastern]**
Gen. Hallucination	**Inform: [Price, Expensive], [Address, 2 G Cambridge Leisure Park Cherry Hinton Road Cherry Hinton]**
Dialogue Response Generator	
Input Materials	Context: ... Customer: It does not matter but please find a guesthouse in the expensive range and include free parking. Assistant: There are no expensive hotels in the area. Could we look at a moderately priced one? Customer: Actually, how about a cheap one? Actions: Request: [Area, ?]
Correct Output	What area of town?
ChatGPT Output	**I have several options for cheap guesthouses with free wifi. Would you like me to provide some recommendations?**
Gen. Hallucination	**I don't have anything in that area. Can I look somewhere else?**

Table 2: Examples with input materials, correct outputs, ChatGPT outputs, and generated hallucination outputs for different modules in task-oriented dialogue systems on the MWOZ 2.1 dataset (Eric et al., 2020; Budzianowski et al., 2018).

Domain detector: The LLM-based domain detector classifies the current user message into a corresponding narrow domain (Hudeček and Dusek, 2023). The narrow domain options are defined in the instruction for the domain detector. Compared to other modules, the domain prediction task has simple input and output structures. However, 6.0% of the samples are still hallucinated. As shown in Table 2, ChatGPT predicts the attraction domain when the user requires a taxi. This example indicates that, even for the simple domain prediction task, we can not avoid the hallucination problem.

Dialogue state tracker: The LLM-based dialogue state tracker extracts slot-value pairs as dialogue states, which represent the user's intentions (Hu et al., 2022b). Slot-value pairs are in the task-specific schema, which is defined by the domain ontology. As shown in Table 2, slot-value pairs from the ChatGPT output are in conflict with the dialogue information. Because the slot-value pairs are used for further database query, hallucinated slot-value pairs result in wrong elements retrieved from the database. Moreover, the hallucination rate of 17.0 % in the dialogue state tracker is much higher than 6.0% from the domain detector, as shown in Table 1. Dialogue state trackers are more likely to generate hallucinations due to their complex task-specific schema.

Dialogue policy: Dialogue policy predicts the assistant actions based on dialogue context, dialogue state, and queried database. Assistant actions include intents, like recommend or inform, and related slot values. The actions will be used for final dialogue response generation. Table 2 shows that the ChatGPT output gives a fabricated restaurant recommendation, and no restaurant information is retrieved from the restaurant domain database. From Table 1, we observed the highest hallucination rate of 28.8% from dialogue policy among the four modules. Assistant actions should be consistent not only with the instruction and dialogue context, but also with the dialogue states and the retrieved database information. The complex input materials lead to a high hallucination rate of predicted actions.

Dialogue response generator: The dialogue response generator generates the assistant response conditioned on the dialogue actions. The assistant response is expected to be informative and task-specific. However, the hallucination example in Table 2 presents that ChatGPT does not map the action to a correct response. Furthermore, we observed a high hallucination rate of 17.6% from the dialogue response generator. This rate indicates that the hallucination problem is also challenging for the dialogue response generator.

5.2 Hallucinated Dataset Generation

Table 2 also presents our generated hallucination outputs for the MWOZ 2.1 dataset. We observed that the generated hallucination output is related to the input and dissimilar to the correct output. This result was achieved by choosing the candidate with the highest hallucination score. The example of the dialogue response generator in Table 2 shows that our generated hallucination is related to the input regarding the topic, and the generated output is dissimilar to the correct output, which ensures that the generated output contains hallucinated content. The examples of different module outputs in Table 2 illustrate the quality achievable with the hallucination score method.

5.3 Hallucination Detection

Table 3 presents the primary hallucination detection results on the MWOZ 2.1 and M2M datasets. The evaluated models include our proposed THD and different LLMs with powerful natural language capabilities.

From Table 3, we observed that our THD achieves better overall performance than other models. We made the following notable findings: (1) Our THD achieves the best overall accuracy performance among evaluated models for two datasets. Compared to ChatGPT, THD shows accuracy values that are higher by 9.83%-46.91% on the MWOZ 2.1 dataset, and 26.30%-66.37% on the M2M dataset. These results indicate that our proposed THD successfully learns the hallucination forms among different modules and domains. (2) Our generated hallucination output dataset is challenging. This is shown by the low hallucination detection accuracy of other models included in the study, and even GPT4 reaches only 80.90%-88.80% on MWOZ 2.1.

Ablation study: To understand the impacts of LoRA in our THD, we conducted an ablation study on the MWOZ 2.1 dataset by removing LoRA. The ablation results in Table 4 show that LoRA improves the performance of THD. Removing LoRA leads to a loss in accuracy of 4.97% for dialogue

	MWOZ 2.1				M2M			
	Domain	State	Action	Response	Domain	State	Action	Response
ChatGPT	47.07	49.20	72.02	54.68	33.38	50.63	-	51.83
Command R	41.75	37.17	68.25	66.83	30.21	68.50	-	59.79
GPT4	86.72	88.80	80.90	82.17	99.58	90.50	-	86.88
THD	93.98	94.42	81.85	85.18	99.75	95.83	-	78.13

Table 3: Primary hallucination detection results with accuracy metric (%) on MWOZ 2.1 and M2M datasets.

	Domain	State	Action	Response
THD	93.98	94.42	81.85	85.18
-LoRA	-	89.45	79.98	84.77

Table 4: Ablation study with accuracy metric (%) by removing LoRA on MWOZ 2.1.

	Domain	State	Action	Response
THD	95.93	82.67	70.13	81.40
GPT4	85.20	81.93	70.73	77.07

Table 5: Accuracy results (%) on 500 collected ChatGPT outputs with human annotations.

states and 1.87% for dialogue actions. These values indicate that LoRA can adapt THD to different narrow domains and enable THD to learn the hallucination forms for the different modules on the MWOZ 2.1 dataset.

Real examples detection: To show the performances in real-life samples, we decided to compare our THD and GPT4 on the 500 ChatGPT outputs that have been annotated during the hallucination situation analysis. Table 5 shows that THD, fine-tuned with generated hallucinations, achieves comparable accuracy performance in real-life samples. This result indicates that THD can benefit from the generated hallucination outputs, which overall simulate the real hallucination situation in task-oriented dialogue modules.

6 Limitation and Future Work

In this paper, we focus on the MWOZ 2.1 and M2M datasets because they are widely used in task-oriented dialogue modules. However, these two datasets cover limited narrow domains and samples, and they contain only English dialogues. The experiments are based on evaluated models, such as the DistilBERT model and ChatGPT, and the described experimental settings. The limited datasets, models, and settings are potentially leading to a bias in the study. In the future, the study framework could be extended to more datasets, different languages, and more developed LLMs, to overcome the domain limitations and reduce the bias.

We highlighted the most vulnerable components of task-oriented dialogue systems based on LLMs, laying the foundations for future engineering improvements to create more reliable virtual assistants. The dialogue policy module needs to be improved for increased reliability. This could be achieved by checking the module output with an accurate and efficient hallucination detector, or by reducing the hallucination rate of the underlying LLM.

7 Conclusion

In conclusion, our paper studies the hallucination problem for all black-box-LLM-based modules in task-oriented dialogue systems with narrow domains. The hallucination situation analysis shows the hallucination rates and forms for all modules, indicating the importance of the hallucination problem. Our dataset generation method, with the hallucination score, successfully simulates the real ChatGPT outputs with hallucinations. Overall, our THD for hallucination detection can benefit from the generated hallucination samples in two datasets. These results encourage future work for hallucination studies in all modules of task-oriented dialogue systems.

Acknowledgments

We thank the reviewers for their important feedback.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, Quyet V. Do, Yan Xu, and Pascale Fung. 2023. A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–718, Nusa Dua, Bali. Association for Computational Linguistics.

Tom B Brown. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Kedi Chen, Qin Chen, Jie Zhou, He Yishen, and Liang He. 2024. DiaHalu: A dialogue-level hallucination evaluation benchmark for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9057–9079, Miami, Florida, USA. Association for Computational Linguistics.

Cohere. 2024. Command r.

Nouha Dziri, Hannah Rashkin, Tal Linzen, and David Reitter. 2022. Evaluating attribution in dialogue systems: The BEGIN benchmark. *Transactions of the Association for Computational Linguistics*, 10:1066–1083.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, Adarsh Kumar, Anuj Goyal, Peter Ku, and Dilek Hakkani-Tur. 2020. Multiwoz 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 422–428.

Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. 2022a. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Yushi Hu, Chia-Hsuan Lee, Tianbao Xie, Tao Yu, Noah A. Smith, and Mari Ostendorf. 2022b. In-context learning for few-shot dialogue state tracking. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2627–2643, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

Vojtěch Hudeček and Ondrej Dusek. 2023. Are large language models all you need for task-oriented dialogue? In *Proceedings of the 24th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 216–228, Prague, Czechia. Association for Computational Linguistics.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781.

Wai-Chung Kwan, Hong-Ru Wang, Hui-Min Wang, and Kam-Fai Wong. 2023. A survey on recent advances and challenges in reinforcement learning methods for task-oriented dialogue policy learning. *Machine Intelligence Research*, 20(3):318–334.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464, Singapore. Association for Computational Linguistics.

OpenAI. 2022. Introducing chatgpt.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020. Adapterhub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020): Systems Demonstrations*, pages 46–54, Online. Association for Computational Linguistics.

Clifton Poth, Hannah Sterz, Indraneil Paul, Sukannya Purkayastha, Leon Engländer, Timo Imhof, Ivan Vulić, Sebastian Ruder, Iryna Gurevych, and Jonas Pfeiffer. 2023. Adapters: A unified library for parameter-efficient and modular transfer learning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 149–160, Singapore. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

V Sanh. 2019. Distilbert, a distilled version of bert: Smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Pararth Shah, Dilek Hakkani-Tür, Gokhan Tür, Abhinav Rastogi, Ankur Bapna, Neha Nayak, and Larry Heck. 2018. Building a conversational agent overnight with dialogue self-play. *arXiv preprint arXiv:1801.04871*.

Khan Sharun, S Amitha Banu, Abhijit M Pawde, Rohit Kumar, Shopnil Akash, Kuldeep Dhama, and Amar Pal. 2023. Chatgpt and artificial hallucinations in stem cell research: assessing the accuracy of generated references–a preliminary study. *Annals of Medicine and Surgery*, 85(10):5275–5278.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *arXiv preprint arXiv:2004.09297*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zheng Zhang, Ryuichi Takanobu, Qi Zhu, MinLie Huang, and XiaoYan Zhu. 2020. Recent advances and challenges in task-oriented dialog systems. *Science China Technological Sciences*, 63(10):2011–2027.

A Appendix

For our hallucination study, we utilize ChatGPT and GPT4 from OpenAI. The Command R model is accessed through the APIs. The "All-mpnet-base-v2" is from Sentence Transformers. The DistilBERT model is from Hugginface (Sanh, 2019; Wolf et al., 2020). The LoRA is implemented with AdapterHub (Poth et al., 2023; Pfeiffer et al., 2020). Because the input length of DistilBERT is limited, we choose the recent utterances as history instead of the whole turns. For the hallucination detection part, we conducted experiments three times for ChatGPT, Command R, and GPT4. The experiments for THD run three times with different seeds. The final accuracy results are the average scores of the three-times experiments.