Towards a Zero-Data, Controllable, Adaptive Dialog System

Dirk Väth, Lindsey Vanderlyn, Ngoc Thang Vu
University of Stuttgart
Stuttgart, Germany
{vaethdk|vanderly|thangvu}@ims.uni-stuttgart.de

Abstract

Conversational Tree Search (Väth et al., 2023) is a recent approach to controllable dialog systems, where domain experts shape the behavior of a Reinforcement Learning agent through a dialog tree. The agent learns to efficiently navigate this tree, while adapting to information needs, e.g., domain familiarity, of different users. However, the need for additional training data hinders deployment in new domains. To address this, we explore approaches to generate this data directly from dialog trees. We improve the original approach, and show that agents trained on synthetic data can achieve comparable dialog success to models trained on human data, both when using a commercial Large Language Model for generation, or when using a smaller open-source model, running on a single GPU. We further demonstrate the scalability of our approach by collecting and testing on two new datasets: *ONBOARD*, a new domain helping foreign residents moving to a new city, and the medical domain *DIAGNOSE*, a subset of Wikipedia articles related to scalp and head symptoms. Finally, we perform human testing, where no statistically significant differences were found in either objective or subjective measures between models trained on human and generated data.

Keywords: Conversational Systems/Dialogue/Chatbots, Corpus, Usability, User Satisfaction

1. Introduction

While the breakthroughs of modern Large Language Models (LLMs) have made the creation of new dialog systems much easier, controlling their generated output remains an open challenge. This makes LLMs especially unsuitable for sensitive domains, e.g., legal or medical domains, where users must be able to implicitly trust the system's output. In such domains, dialog designers usually have the choice between implementing an FAQ-retrieval system or a hand-crafted dialog system.

FAQ systems directly match user queries to question/answer pairs curated by domain experts, allowing close control of outputted texts (Wu et al., 2005). However, as they are single-turn systems and cannot ask clarifying questions, they are only able provide general answers, rather than personalized content for a specific user and their situation. Including information for multiple cases in one answer would make them unapproachably long, while adding FAQs for each case, would make retrieval challenging. Retrieval accuracy itself is an open challenge (Thakur et al., 2021), creating a trade-off: Either providing a single, possibly incorrect answer to a user's question, or providing multiple answers and shifting the burden of selecting the correct one to the user, which might be challenging for users unfamiliar with the domain.

Dialog systems, in contrast, allow for turn-based interactions, which can provide shorter, personalized answers, as well as support users new to a domain without enough experience to formulate precise questions. However, such systems either suffer from longer interactions (for handcrafted systems), or require large amounts of training data (Raghu et al., 2021) and lack transparency and controllability (Gao et al., 2018) (in the case of machine learning approaches), making them less suitable for low-resource settings (Zhang et al., 2020) or sensitive domains (Cohen, 2020).

Väth et al. (2023) address this problem by proposing a new type of hybrid dialog task bridging these two interaction styles, called Conversational Tree Search (CTS). In this task, dialog experts first define a dialog tree. An agent then learns to either walk the user through each node in the tree, or to skip over parts not required to answer a user's more specific question. In this way, the agent is able to adapt its behavior to the user's preferred interaction style, supporting both specific and vague user queries, without sacrificing the controllability required in sensitive domains.

However, CTS still requires that dialog designers collect a corpus of real-user utterances, which poses a barrier to scaling this approach to new domains, especially for large and complicated domains. The goal of this paper is to remove this barrier by exploring how CTS can scale to new domains through the use of synthetically generated training data.

Concretely, we seek to answer the following research questions:

- **(RQ1)** How can we effectively generate data for a zero data approach to training CTS agents?
  - (**RQ1.1**) How can we analyze the quality of generated data?

LREC-COLING 2024, pages 16433–16449

- (**RQ1.2**) How do agents trained on generated data perform in simulation, compared to agents trained on human data?
- (**RQ1.3**) How well do the data generation techniques transfer to new domains?

- (**RQ2**) How does a CTS agent trained on generated data perform with real users compared to an agent trained on human data?

To address these questions, we investigate how LLMs can be leveraged to automatically generate training data for new domains, while at the same maintaining the controllability aspect of the CTS task. We compare the quality of different data generation schemes by evaluating the performance of Reinforcement Learning (RL) agents trained on the synthetic data. Then, we test scalability of our approach to new domains in simulation using multiple generative LLMs. Finally, we perform user testing to verify the transferability to real-world use cases. All code and data is publicly available.[1]

Our main contributions are: 1) Creating two new datasets, *ONBOARD* and *DIAGNOSE*. 2) Improving the training procedure for the CTS agent, increasing absolute dialog success by more than 18%. 3) Introducing a new prompting method for generating diverse data, and demonstrating that automatic diversity and answerability metrics can provide insights for downstream dialog performance. 4) Demonstrating that our generation techniques scale to new domains, where agents trained on synthetic data show comparable (no statistically significant difference) or better dialog success than agents trained on human data. 5) Showing that success of agents in simulation translates to successful interactions with real users, with no statistically significant differences.

2. Related Work

2.1. Task-oriented Dialog Systems

While open-domain dialog systems allow users to freely talk about any topic without a concrete goal, task-oriented dialog systems focus on helping a user reach a specific goal. Many task-oriented dialog systems use a slot-filling approach, where the dialog system tries to fill values for a selection of slots, e.g. cuisine type, that are necessary to reach that goal from the user (Bobrow et al., 1977). While slot filling approaches can allow hand-crafted dialog policies to follow pre-defined dialog flows (Lucas, 2000), or can help efficiently narrowing down searches across e.g. database rows, such as finding restaurants or getting trip recommendations (Louvan and Magnini, 2020), they are usually unable to perform semantic searches over the dialog domain and in cases of learned systems, unable to follow a dialog-designer controlled flow.

2.2. Adaptive Dialog Systems

Research into adaptive dialog systems aims to better align dialog system output with user expectations. Much research in this area uses generative models to adapt linguistic style, e.g., adjusting utterances depending on users' emotional states (Ma et al., 2020) or personalities (Yang et al., 2018; Firdaus et al., 2023). However, generative models are by their nature difficult to control (Dušek and Kasner, 2020). Some approaches even adapt the complexity of language (Janarthanam and Lemon, 2014). In order to adapt underlying system behavior, however, additional cues have usually been required, e.g. social cues like laughter (Ritschel and André, 2018), or explicit fine-tuning by the user (Chen and Pu, 2012; Narducci et al., 2018). However, eliciting such social cues is difficult for text-based systems and asking for explicit feedback places extra burden on the user.

2.3. Controllable Dialog Systems

In sensitive domains, it is crucial subject-experts maintain control of dialog flow to ensure correctness of system outputs. However, purely hand-crafted systems struggle to handle the breadth of possible user inputs. To this end, several hybrid approaches have been investigated. Early approaches involved hand-crafting the set of actions allowed at a given dialog turn (Williams, 2008). More recent approaches expand on this idea for neural systems (Williams et al., 2017; Liang and Yang, 2018; Razumovskaia and Eskenazi, 2019), where the action space can be constrained using masks, e.g., by automatically converting expert designed dialog trees into hybrid code networks (Shukla et al., 2020). While such approaches help control dialog agent behavior, they do not provide a mechanism for skipping portions of a dialog irrelevant to a user, which leads to longer interactions that can be frustrating for users with more domain familiarity.

2.4. Data Generation and Augmentation

Common data augmentation approaches include lexical substitution (Wei and Zou, 2019), where tokens are inserted, deleted or substituted with semantically similar replacements, as well as back-translation (Sennrich et al., 2016) where data is

[1] https://github.com/DigitalPhonetics/conversational-tree-search/tree/generated_v3

automatically translated into other languages before being translated back to the source language. While such approaches can help to expand an existing dataset, they still require seed data, which may not exist for new domains.

To address this, research in, e.g., the field of low-resource Question Answering (QA) has started exploring the role of LLMs in data generation (Puri et al., 2020; Chen et al., 2023). Given a text, LLMs can be prompted to generate questions about it, e.g., by asking the model to generate a question for which a given named entity is the answer (Li et al., 2023).

However, LLMs are black-box algorithms and suffer from hallucination (Azaria and Mitchell, 2023; Peng et al., 2023; Manakul et al., 2023). As such, it is difficult to guarantee that the generated questions are logical, natural, or answerable by the original text. Moreover, commonly used automatic evaluation metrics for text generation do not necessarily correlate with human judgment (Nema and Khapra, 2018). In light of this, we explore different generation strategies and techniques for analyzing the artificial data quality, rather than trusting a single metric.

A recent approach in the dialog community trains a model for generating synthetic dialog acts and user utterances for flowchart-grounded troubleshooting dialogs (Zhan et al., 2023). While this method also relies on the domain representation in form of a structured graph, our generation approach does not require any model training, nor any training data besides the domain graph itself. Additionally, CTS is not limited to the specific task format of trouble-shooting dialogs.

2.5. Conversational Tree Search

The goal of CTS, as outlined by Väth et al. (2023), is to train an RL agent to traverse a dialog tree, guiding a user to the answer for a given question. By using fixed system outputs (which can be personalized via a template mechanism), and by preventing skipping between branches of the dialog tree, the CTS task allows subject-experts to maintain controllability.

At the same time, the trained agent can adapt its behavior to different interaction styles, based on the users' utterances. CTS proposes two subtasks: guided mode and free mode, representing the extreme cases of information seeking scenarios, as well as the interpolation between. Guided mode supports users unable to formulate their information need as a specific question, by guiding them step-by-step through each node in the dialog graph (e.g., new users not familiar with a domain). In contrast, free mode aims to support users with a specific question by learning to skip over as many nodes as possible, while still clarifying the information need enough to deliver an appropriate and personalized answer. Figure 1 shows three example dialogs for the same user goal, and how a CTS agent would adapt to each scenario, deciding to output or skip nodes as needed.

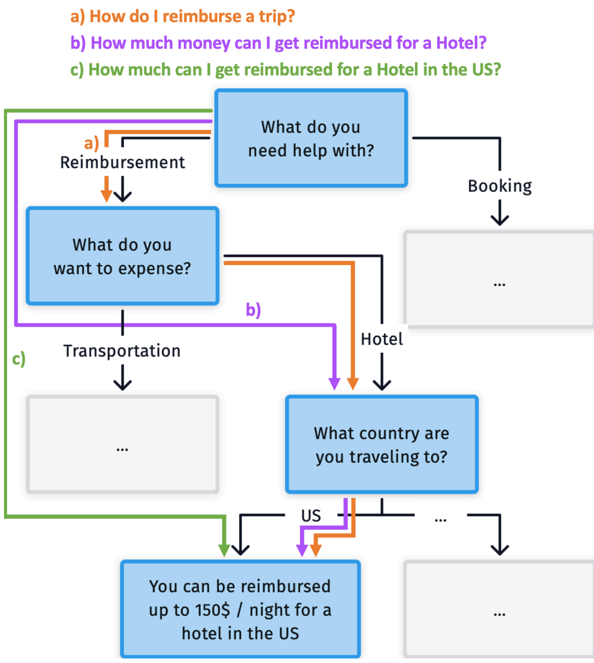


Figure 1: Example of the CTS agent adapting its behavior based on the information content of the initial user utterance (Väth et al., 2023).

Training is performed against a simulated user, which represents the RL environment. For each simulated dialog, a random goal node is drawn which the simulated user is trying to reach, by asking questions or responding to system requests.

3. Datasets

To investigate the scalability of our data generation techniques, we examine the performance of the CTS agent on three new datasets, and compare to the original *REIMBURSE* dataset from Väth et al. (2023). In contrast to the *REIMBURSE* dataset, the goal of all new datasets is to serve as a zero-data test-bed for testing training and testing models on data generated directly from the nodes themselves. While we do provide a test and a train set, like that in *REIMBURSE*, the goal of this is to allow for the training of reference models to act as a benchmark for models trained entirely on generated data.

3.1. REIMBURSE

The *REIMBURSE* dataset as proposed by Väth et al. (2023) is a German language dataset for the

Dataset	Split	#Nodes	Tree Depth	Max. Node Degree	#User Questions	Avg. User Questions	#Answer Paraphrases	Avg. Answer Paraphrases
REIMBURSE	Train Test	123	32	14	279 173	3.5 2.2	246 162	3.4 2.2
REIMBURSE-En	Train Test	123	32	14	279 173	3.5 2.2	246 162	3.4 2.2
DIAGNOSE **DIAGNOSE**	Train Test	98	10	6	219 150	2.9 2.0	298 298	3.0 3.0
ONBOARD **ONBOARD**	Train Test	88	15	9	141 117	2.4 2.0	175 152	3.1 2.7

Table 1: Overview of original *REIMBURSE*, translated *REIMBURSE-En*, and newly created *ONBOARD* and *DIAGNOSE* datasets (numbers rounded to one decimal).

CTS task. It is a challenging real-world dataset in the travel reimbursement domain, created with domain experts. Along with the dialog tree, questions and answer paraphrases were collected from real user interactions. These questions and answer-paraphrases have been split into a train and test set which can each be used by the provided user simulator to generate an arbitrary number of simulated dialogs. A breakdown of the dataset statistics can be found in Table 1.

Although we do not train any new models on this dataset, we use it as a benchmark to compare the performance of our agents to.

3.2. REIMBURSE-En

In order to make the CTS task more accessible to a wider audience, we choose to translate the *REIMBURSE* dataset to English. Additionally, this opens up more options for language models and resources, which might not have been available for the original German data. This dataset represents a direct translation of the *REIMBURSE* dataset, sharing all of the same characteristics, in order to allow for comparisons to the findings of the original CTS paper. The translation was performed manually by a bilingual domain-expert in order to obtain a faithful and factually correct English equivalent. Dataset statistics are shown in Table 1.

3.3. DIAGNOSE

The *DIAGNOSE* dataset was created for the medical domain. It was designed to help users identify different medical conditions based on symptoms, as well as to find out more about treatment options and risk factors. The dataset is based on a small subset of Wikipedia articles about conditions related to scalp and head symptoms. *DIAGNOSE* was designed to be comparatively easy. Even though the node texts contain a large amount of domain-specific vocabulary, the dialog tree has a lower maximum node degree and a shallower tree depth than *REIMBURSE-En*. Additionally, the dialog graph for this domain does not contain any variable- or logic nodes. A breakdown of dataset properties can be found in Table 1.

An example node and associated questions can be seen below:

NODE TEXT: Anemia symptoms include fatigue, pale skin and gums, blue color in the whites of the eyes, brittle nails, irritability, dizziness, sore tongue, shortness of breath, unusual food cravings, and headache.

QUESTION 1: What are symptoms of anemia?

QUESTION 2: How do I know if I have anemia?

QUESTION 3: Is a sore tongue a common symptom of anemia?

3.4. ONBOARD

The *ONBOARD* dataset provides users with information about moving to a new city in a foreign country, and the legal and financial steps they will need to undertake, i.e., setting up bank accounts, acquiring health insurance, applying for required visas or residence permits, etc. This domain presents an additional challenge as it contains code-switching for topics related to legal issues, in order provide users with official names for documents, concepts, and institutions. Similar to the *REIMBURSE* dataset, the dialog tree for *ONBOARD* contains multiple variable nodes and several logic nodes. A breakdown of the dataset statistics can be found in Table 1.

An example of a a dialog node and test questions is given below.

NODE TEXT: The registration office will provide you with a confirmation of your registration [Meldebestätigung], which you will need for opening a bank account and for obtaining a residence permit (if applicable).

QUESTION 1: Where do I get confirmation that I've registered my address?

QUESTION 2: What do I need the confirmation of registering my address for?

4. Dialog Agent Implementation

For our RL dialog agent, we follow the architecture and training process outlined in (Väth et al., 2023) with the following changes:

1) We swap the original language model for an MPNET (Song et al., 2020) based Sentence-Transformer (Reimers and Gurevych, 2019), as the new datasets we introduce are in English, and it reports the highest average performance of pre-trained Sentence-Transformers for English.

2) In contrast to free mode, rewards for guided mode only considered whether the agent moved to the correct next node, rather than checking that a global goal was reached by the end of the dialog. After analyzing conversations between CTS agent and user simulator obtained by the original implementation, we believe it is more realistic that, even in guided mode, users would have a consistent question they wanted answered. Therefore, we now draw global goals for guided mode users (a node anywhere in the graph) instead of choosing one of the immediate neighboring nodes as the next goal each turn. We then assign a large reward to reaching the global goal. At the same time, we keep a small positive reward for skipping to the correct follow-up node along the sampled trajectory, as a sequence of locally correct decisions (reaching a correct immediate neighbor) implies global correctness (reaching the correct goal node). These changes result in a harsher evaluation metric for dialog success, since e.g. in a 5-step dialog, following a correct trajectory, but missing the final goal in the last turn, will now result in a failed dialog (0% success) instead of a partially successful dialog (80% success), which we consider to be more realistic.

3) Finally, the original CTS agent was trained jointly on navigating the graph and on predicting the appropriate interaction style (intent). Here, we scale the loss of the interaction style prediction objective down to 0.1 to emphasize learning Q-values as the main task: $\mathcal{L} = \mathcal{L}_{\text{ddqn}} + 0.1\mathcal{L}_{\text{intent}}$. We found this had no significant impact on the interaction style prediction F1 score.

4) We tune several other hyperparameters, increasing the batch size from 128 to 256, and the training steps from $1.5e6$ to $2e6$.

All hyperparameters for training the dialog agent are listed in Appendix A.

5. Data Generation Methods

As the user simulator from Väth et al. (2023) requires both, initial user questions and per-node user responses, we explore methods for generating both of these types of utterances. We test these generation methods with a small LLM, and with a large commercial one, both of which can process separate system and user input directives.

5.1. Question Generation

Method 1 The first method, Gen$_{V1}$, is a naive prompt instructing an LLM to generate diverse, FAQ-style questions about a given dialog node's text via the system directive. The amount of questions to generate and the node context are then given via user input (see Table 2).

Method 2 For Gen$_{V2}$, we use the same user input, but change the system directive to explicitly generate shorter questions (Table 2).

Method 3 For the last method, Gen$_{V3}$, we were inspired by Li et al. (2023) and Chen et al. (2023), who use Named Entity Recognition (NER) to steer question generation. However, these approaches only generate cloze questions, where the named entity is the answer, severely limiting the diversity of generated questions (Puri et al., 2020). Therefore, we develop a novel mixed method to increase question diversity. We first generate 3 questions about the whole node text using the Method 2, to get a basic coverage of the node. Then, we perform NER and explicitly prompt the LLM to generate three questions about each entity –instead of forcing the entities to only be the answer– using a second set of prompts (see Table 2). If the total number of generated questions is lower than 10, we generate the difference using Method 2.

5.2. Response Generation

To generate responses, we extract all nodes requiring user input from the dialog graph. Then, we instruct the LLMs to generate 5 paraphrases for each possible answer prototype, in the context of the full node text (Table 3; A). Additionally, to mimic different user interaction styles, we instruct the LLMs to generate 5 paraphrases of the the responses using only keywords (Table 3; B).

6. Experimental Setup

6.1. RQ 1.1: Analysis of Generated Data

We generate data using the methods described in sections 5.1 and 5.2. We use two different LLMs: ChatGPT (gpt-3.5-turbo, via API) [2] and a LLAMA-based (Touvron et al., 2023), instruction fine-tuned and quantized model [3] that fits onto a

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

[3] https://huggingface.co/TheBloke/upstage-llama-30b-instruct-2048-GPTQ

Method	Role	Context	Prompt
V1	System	Node text	You are a truthful assistant, generating diverse FAQ-style questions given some facts. The generated questions should be answerable using the given fact only, without additional knowledge. The questions should also be human-like. Try to vary the amount of information between questions. Present the results in a numbered list.
	User	Node Text	Generate 10 FAQ-style questions about the given facts: "{NODE TEXT}".
V2	System	Node Text	You are a truthful assistant, generating diverse FAQ-style questions given some facts. The generated questions should be answerable using the given fact only, without additional knowledge. The questions should also be short and human-like. Try to vary the amount of information between questions. Present the results in a numbered list.
	User	Node Text	(same as V1)
V3	System	Node Text	(same as V2)
	User	Node Text, NER	Generate 3 questions about the entity "{NER}" from the fact: "{NODE TEXT}"

Table 2: Prompt templates for generating synthetic question data.

Method	Role	Context	Prompt
A	System	Node text	You are generating semantically similar paraphrases for a given response to some question. The generated response paraphrases should be human-like and short, using frequently used words and phrases only. Present the results in a numbered list.
	User	Node Text	Generate 5 paraphrases for the response "{RESPONSE TEXT}" to the question "{NODE TEXT}"
B	System	Node Text	You are shortening a given response to some question into a keyword-like prompt. Present the results in a numbered list.
	User	Node Text, NER	Generate 5 options for shortening the response "{RESPONSE TEXT}" to the question "{NODE TEXT}"

Table 3: Example of prompting method for generating synthetic user response data.

single NVIDIA GeForce RTX 3090 graphics card. Gen$_{V3}$ uses Stanza (Qi et al., 2020) for NER.

To calculate question similarity, we use the Sentence-Transformer model from section 4. Answer confidence scores are calculated with a QA model [4] pretrained on the SQUAD2.0 dataset (Rajpurkar et al., 2018), using a generated question and associated node text that is supposed to contain the answer as inputs. Finally, we measure diversity using Self-BLEU (Zhu et al., 2018) scores.

6.2. RQ 1.2: Human Data vs. Synthetic Data

For automatic evaluation, we use the updated CTS user simulator (section 4) with 500 randomly chosen dialog goals on the *REIMBURSE-En* test split. We evaluate not only the combined success rate (average between guided and free mode success), but also present a metric representing the user's *perceived dialog length*, which counts only the nodes shown to the user.

6.3. RQ 1.3: Method Generalizability

To evaluate how well our data generation method generalizes to new domains, we perform additional evaluation in simulation, analogous to (section 6.2), using the test splits of the new datasets *ONBOARD* and *DIAGNOSE*.

[4]https://huggingface.co/deepset/roberta-large-squad2

6.4. Human Evaluation (RQ 2)

To understand how performance of an agent trained on generated data translates to real-world users, we recruit 44 participants from the crowdsourcing platform Prolific[5] to take part in human evaluation. Participants were native English speakers with varying experience with business travel (self-rating between 2 and 5 on a 5 point Likert-scale). They were compensated at the platform recommended rate of 9£/hour. The experiment took roughly 20 minutes.

Study Design We asked each participant to interact with either a CTS agent trained on real data or one trained on generated data in the *REIMBURSE* domain. Apart from demographic information, we ask for previous experience with dialog systems and with business travel. During the experiment, participants were asked to complete three conversations with their assigned dialog system. Each conversation, they were randomly assigned a new goal, covering one of three expected interaction styles: 1) "open" goals representing a general/vague information need, 2) "easy" goals representing a concrete information need, and 3) "hard" goals representing a concrete information need requiring personalized information to correctly answer. Personalized information refers to the user's specific circumstances, e.g. trip duration or funding organization, which can change the dialog flow. Between each dialog, users were

[5]https://www.prolific.com

asked to rate their subjective perception of dialog length and how well their question was answered. After the interaction, they were asked rate the usability of the dialog agent, how much they trusted it, and its reliability. For more details see Appendix B.

6.4.1. Evaluation Metrics

The perceived dialog length was measured on a 5-point scale from 1 (much too short) to 5 (much too long). Perceived success was measured on a 4-point scale, where users were asked to rate how well their question had been answered from 1 (not at all) to 4 (completely). Additionally, the objective dialog length and success condition were logged for each dialog. Usability of the dialog agent was measured using the Universal Measure of User Experience scale developed by Finstad (2010). User trust was measured using the reliability and trust subscales from Körber (2018)).

7. Results & Discussion

Before testing performance of agents trained on generated data, we first verify our changes to the CTS agent. As the hyperparameters for the original agent were tuned on the German dataset, for fairness, we report the original CTS agent's performance on both English and German (Table 5).

Our changes to the CTS agent improve the combined success rate by over 10% compared to the original agent on the German *REIMBURSE* dataset and 18% for the English *REIMBURSE-En*. It should be noted that the actual improvement over the German agent is likely larger, as the success metric reported for German comes from (Väth et al., 2023), rather than the new and harsher metric we use for English (section 4).

7.1. RQ 1: Transitioning to a Zero Data Approach

RQ 1.1 Analyzing the quality of generated data
Looking at the question lengths between human data and data generated by *GEN*$_{V1}$ (Figure 2), we observe that the generated questions seem to be longer than human questions. When manually inspecting the generated questions, we also find them to be much less natural than those from the human data.

We amend the original prompt used, creating *GEN*$_{V2}$, to explicitly ask for short outputs (subsection 5.1) in an effort to align the syntax of the generations better with the human data. This change to the prompt shifts the distribution of question lengths more towards the human training distribution, and qualitatively yields more natural utterances. However, it still does not ensure that the artificial data is semantically similar to human data.

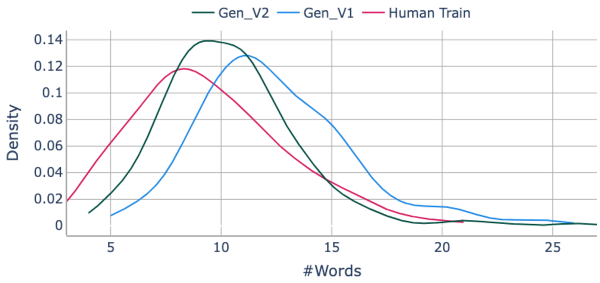


Figure 2: Smoothed density plot of question lengths from human data and generated data.

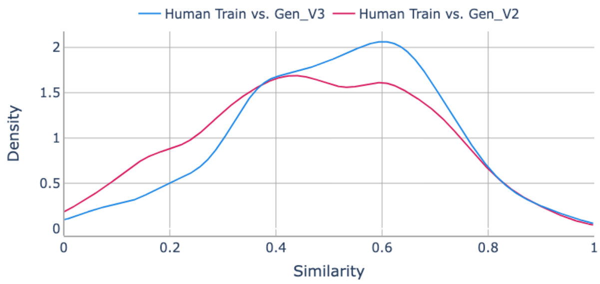


Figure 3: Smoothed density plot of question similarities between human generated training data.

To investigate how semantically similar the generated questions are to human data, we calculate the pair-wise similarities between all human and generated questions for each node from the dialog graph, and then average the similarities across all nodes (Figure 3). Here, we see that the Gen$_{V2}$ data is still quite distinct from the human data.

When manually inspecting the generations, we find that generated questions tend to focus only on one part of the node text, making them lack diversity and omit topics real users might ask about. To address this, we develop the novel two-step Gen$_{V3}$ prompt, steering the model to explicitly ask about all named entities in a node (subsection 5.1). We see that doing so significantly ($p < 1.82e-11$) increases the similarity of the generated (avg.: 0.52) to the human training data than Gen$_{V2}$ (avg.: 0.47), as measured with a standard t-test.

We also look at the diversity of the generated questions. The self-BLEU scores (Table 4) show that the Gen$_{V3}$ data are the most diverse. This metric can be used to analyze the quality of the generated data even in the absence of human comparison data.

Training Data	n-1	n-2	n-3	n-4	n-5
Human	**0.78**	**0.68**	**0.60**	**0.54**	**0.49**
V1	0.95	0.92	0.87	0.83	0.80
V2	0.95	0.90	0.85	0.80	0.76
V3	**0.85**	**0.78**	**0.71**	**0.66**	**0.62**

Table 4: Self-BLEU scores for different n-gram sizes on human and generated data.

Model	Training Data	Avg. Perceived Length (guided)	Avg. Perceived Length (free)	Success (combined)	Dialog Mode Prediction F1	Dialog Mode Prediction Consistency
Original	human (GER)	n/a	n/a	62.58%	0.85	**1.0**
Original	human (EN)	n/a	n/a	55.28%	0.86	0.87
Ours	human (EN)	13.56	2.95	**73.86%**	**0.94**	0.96
Ours	V1 (LLAMA)	13.53	3.41	64.17%	**0.98**	**0.97**
Ours	V2 (LLAMA)	11.71	3.65	65.02%	**0.98**	0.95
Ours	V3 (LLAMA)	12.89	3.45	**69.44%**	0.96	0.95
Ours	V1 (ChatGPT)	13.02	3.65	64.35%	**0.98**	**0.97**
Ours	V2 (ChatGPT)	14.55	3.71	66.67%	0.95	**0.97**
Ours	V3 (ChatGPT)	12.87	3.59	**68.41%**	**0.98**	**0.97**

Table 5: Simulation results on *REIMBURSE(-En)* test splits of original CTS agent (German), our improved agent (English), and our CTS agent trained on generated data only (English).

In conjunction with diversity, we estimate the average "answerability" via QA confidence scores of the generated questions, given the node text as answer. Here, we also see that the improvements from Gen$_{V3}$ and Gen$_{V2}$ together also significantly ($p < 0.0003$) increase the average answerability, from an average of 0.36 with naive prompt to 0.42 with Gen$_{V3}$, according to a t-test.

When looking at downstream performance (Table 5), we see that improvements in these metrics also lead to higher dialog success, suggesting they can be used as an indicator of generation quality.

RQ1.2: Human Data vs. Synthetic Data To investigate whether synthetic data can be a viable alternative to human data, we compare agent performance in simulation. From Table 5, we see that the best performing agent trained on artificial data (Gen$_{V3}$: 69.44% success) performs comparably to the best performing agent trained on human data (CTS$_{ours}$: 73.86% success). Using a standard t-test, we find no statistically significant difference.

RQ1.3: Generalizing to new domains To test of the scalability of our generation methods, we analyze model performance on two new domains. As each of these has their own challenges (section 3), we compare each model trained on generated data to a baseline trained on human data.

When looking at Table 6, the agent trained on data generated by LLAMA is again nearly able to match the performance of the model trained on human data for the *DIAGNOSE* dataset, while the model trained on data generated by ChatGPT surpasses it. On the other hand, the *ONBOARD* dataset may present a more challenging domain, due in part to the code-switching present in the dialog nodes. Despite this, the model trained on data from ChatGPT nearly reaches the performance of models trained on human data.

Based on this, we find that the generation techniques do appear to scale to new domains, as t-tests show no statistically significant differences between the best synthetically trained agents and the agents trained on real data in any domain.

7.2. RQ 2: Human Evaluation

7.2.1. Generated vs. Real Data

After performing human evaluation, we find that there are no statistically significant differences (using a standard t-test) between either subjective or objective measures of success or dialog length (Table 7). Additionally, we find no difference in the reported trust, reliability, or usability scores between either group. This suggests that there is no human-observable loss in performance when using generated data compared to real data, either in terms of objective metrics or subjective metrics.

7.2.2. Human Evaluation vs. Simulator

Finally, to validate our updated user simulator, we additionally compare the objective performance metrics from the human evaluation (Table 7) to those obtained in simulation (Table 5). We find that the success rates between the simulated and human dialogs are very comparable (73.86% and 77.59% respectively for the model trained on human data, and 69.44% and 72.73% for the model trained on generated data). We perform statistical analysis using Welch's t-test to account for the difference in sample size, and find no significant difference, regardless of the source of training data.

Based on this, we conclude that results from simulation translate well to real human interaction, suggesting the simulator can be a good proxy for real user evaluation. We therefore expect the results reported in (Table 6) will translate to similar performance with real users.

8. Conclusion

In this paper, we present two new and publicly available datasets, *ONBOARD*, providing help for moving to a new city in a foreign country, and *DIAGNOSE*, a medical domain. The datasets each consist of a dialog tree and human-collected text inputs.

We apply a harsher, more realistic evaluation metric and improve on the agent training method

Domain	Training Data	Avg. Perceived Length (guided)	Avg. Perceived Length (free)	Success (combined)
DIAGNOSE	human	6.42	2.29	76.31%
DIAGNOSE	V3 (LLAMA)	6.62	2.95	71.08%
DIAGNOSE	V3 (ChatGPT)	5.65	2.46	**85.12%**
ONBOARD	human	7.88	2.98	**73.61%**
ONBOARD	V3 (LLAMA)	7.91	3.52	63.38%
ONBOARD	V3 (ChatGPT)	7.60	3.58	70.72%

Table 6: Performance of CTS agents trained on human and generated data on the new domains *DIAGNOSE* and *ONBOARD* in simulation.

Training Data	# Turns	Success	Perceived Length	Answer Satisfaction
Human	6.14	77.59	2.88	2.93
V3	5.27	72.73	2.65	2.73

Table 7: Average objective and subjective performance metrics of a CTS agent trained on human data vs. generated data.

from the original CTS (Väth et al., 2023), increasing dialog success by over 18%.

Given a dialog tree, we explore several zero-data prompting-based methods for generating user utterance data to train a CTS agent, developing a novel two-stage prompting approach to increase question diversity. Through this process, we find that automatic scores for diversity and answerability can be indicative of downstream dialog task performance.

Furthermore, we show that there is no statistically significant difference in objective metrics between agents trained on human data or on generated data in the *REIMBURSE-En* domain. We verify this both through simulation and through testing with real users. User evaluation further reveals no statistically significant differences on subjective metrics (trust, reliability, usability, subjective length, or subjective dialog success) either. This suggests that we can effectively generate training data from a dialog tree, such that CTS agents can be trained in zero data settings with negligible performance loss. We also find that the size of the tested LLMs does not result in significant differences in task performance.

To evaluate how well our techniques scale to new domains, we further tested agent performance on both new datasets we introduced. For *ONBOARD*, we again find that performance of agents trained on generated data is comparable to that of agents trained on human data. For *DIAGNOSE*, performance can even exceed that of the agent trained on human data. This suggests that our methods scale well to new domains.

9. Ethical Considerations

To ensure that users could give informed consent, we provided a detailed description of the task and research objectives both on the crowdsourcing platform and once they had accepted the task. In respect of participant privacy, we specifically did not collect personally identifying data from any users. To this end, we store all logs and survey responses using an anonymous hash generated based on a given username, rather than with the username itself. In this way, users could log in again if they needed to take a break in the middle of the interaction, but we had no way of directly linking any recorded results to, e.g., users' Prolific account identifiers. To ensure that participants were fairly compensated, we followed best practices recommended by the crowdsourcing platform paying users at 9£/hr. We additionally used our pilot study to verify that our estimated time was below the median time we selected when advertising the task.

10. Limitations

While we try to cover many different real-world use cases with the presented domains, we cannot account for the challenges of all possible future domains. Additionally, although our work removes the necessity to collect training data, creating a dialog tree is still required (which may be large for complex domains). Finally, replicating the exact data generated and analyzed in this paper depends on the specific versions of the LLMs used.

11. Bibliographical References

Amos Azaria and Tom Mitchell. 2023. The internal state of an llm knows when its lying. *arXiv preprint arXiv:2304.13734*.

Daniel G. Bobrow, Ronald M. Kaplan, Martin Kay, Donald A. Norman, Henry Thompson, and Terry Winograd. 1977. Gus, a frame-driven dialog system. *Artificial Intelligence*.

Li Chen and Pearl Pu. 2012. Critiquing-based recommenders: survey and emerging trends. *User Modeling and User-Adapted Interaction*, 22(1):125–150.

Xiusi Chen, Yu Zhang, Jinliang Deng, Jyun-Yu Jiang, and Wei Wang. 2023. Gotta: generative few-shot question answering by prompt-based cloze data augmentation. In *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*, pages 909–917. SIAM.

Philip R. Cohen. 2020. Back to the future for dialogue research. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 13514–13519. AAAI Press.

Ondřej Dušek and Zdeněk Kasner. 2020. Evaluating semantic accuracy of data-to-text generation with natural language inference. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 131–137, Dublin, Ireland. Association for Computational Linguistics.

Kraig Finstad. 2010. The usability metric for user experience. *Interacting with computers*, 22(5):323–327.

Mauajama Firdaus, Arunav Shandilya, Asif Ekbal, and Pushpak Bhattacharyya. 2023. Being polite: Modeling politeness variation in a personalized dialog agent. *IEEE Transactions on Computational Social Systems*, 10(4):1455–1464.

Jianfeng Gao, Michel Galley, and Lihong Li. 2018. Neural approaches to conversational ai. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 1371–1374, New York, NY, USA. Association for Computing Machinery.

Srinivasan Janarthanam and Oliver Lemon. 2014. Adaptive generation in dialogue systems using dynamic user modeling. *Computational Linguistics*, 40(4):883–920.

Moritz Körber. 2018. Theoretical considerations and development of a questionnaire to measure trust in automation. In *Congress of the International Ergonomics Association*, pages 13–30. Springer.

Junlong Li, Zhuosheng Zhang, and Hai Zhao. 2023. Self-prompting large language models for zero-shot open-domain qa.

Weiri Liang and Meng Yang. 2018. Hierarchical hybrid code networks for task-oriented dialogue. In *Intelligent Computing Theories and Application - 14th International Conference, ICIC 2018, Wuhan, China, August 15-18, 2018, Proceedings, Part II*, volume 10955 of *Lecture Notes in Computer Science*, pages 194–204. Springer.

Samuel Louvan and Bernardo Magnini. 2020. Recent neural methods on slot filling and intent classification for task-oriented dialogue systems: A survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 480–496, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Bruce Lucas. 2000. Voicexml for web-based distributed conversational applications. *Communications of the ACM*, 43(9):53–57.

Yukun Ma, Khanh Linh Nguyen, Frank Z. Xing, and Erik Cambria. 2020. A survey on empathetic dialogue systems. *Information Fusion*, 64:50–70.

Potsawee Manakul, Adian Liusie, and Mark JF Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. *arXiv preprint arXiv:2303.08896*.

Fedelucio Narducci, Marco de Gemmis, Pasquale Lops, and Giovanni Semeraro. 2018. Improving the user experience with a conversational recommender system. In *International Conference of the Italian Association for Artificial Intelligence*, pages 528–538. Springer.

Preksha Nema and Mitesh M. Khapra. 2018. Towards a better metric for evaluating question generation systems. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3950–3959, Brussels, Belgium. Association for Computational Linguistics.

Baolin Peng, Michel Galley, Pengcheng He, Hao Cheng, Yujia Xie, Yu Hu, Qiuyuan Huang, Lars Liden, Zhou Yu, Weizhu Chen, et al. 2023. Check your facts and try again: Improving large language models with external knowledge and automated feedback. *arXiv preprint arXiv:2302.12813*.

Raul Puri, Ryan Spring, Mohammad Shoeybi, Mostofa Patwary, and Bryan Catanzaro. 2020. Training question answering models from synthetic data. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5811–5826, Online. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A Python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*.

Dinesh Raghu, Shantanu Agarwal, Sachindra Joshi, and Mausam. 2021. End-to-end learning of flowchart grounded task-oriented dialogs. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4348–4366, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Evgeniia Razumovskaia and Maxine Eskenazi. 2019. Incorporating rules into end-to-end dialog systems. In *Proc. 3rd NeurIPS Workshop on Conversational AI, Vancouver, Canada*, pages 1–11.

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 and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Hannes Ritschel and Elisabeth André. 2018. Shaping a social robot's humor with natural language generation and socially-aware reinforcement learning. In *Proceedings of the workshop on NLG for human–robot interaction*, pages 12–16.

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.

Swadheen Shukla, Lars Liden, Shahin Shayandeh, Eslam Kamal, Jinchao Li, Matt Mazzola, Thomas Park, Baolin Peng, and Jianfeng Gao. 2020. Conversation Learner - a machine teaching tool for building dialog managers for task-oriented dialog systems. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 343–349, Online. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in Neural Information Processing Systems*, 33:16857–16867.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. In *35th Conference on Neural Information Processing Systems (NeurIPS 2021) Track on Datasets and Benchmarks*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Dirk Väth, Lindsey Vanderlyn, and Ngoc Thang Vu. 2023. Conversational tree search: A new hybrid dialog task. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1264–1280, Dubrovnik, Croatia. Association for Computational Linguistics.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Jason D Williams. 2008. Integrating expert knowledge into pomdp optimization for spoken dialog systems. In *Proceedings of the AAAI-08 Workshop on Advancements in POMDP Solvers*, volume 2, page 25.

Jason D. Williams, Kavosh Asadi, and Geoffrey Zweig. 2017. Hybrid code networks: practical and efficient end-to-end dialog control with supervised and reinforcement learning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 665–677, Vancouver, Canada. Association for Computational Linguistics.

Chung-Hsien Wu, Jui-Feng Yeh, and Ming-Jun Chen. 2005. Domain-specific faq retrieval using independent aspects. *ACM Transactions on Asian Language Information Processing*, 4(1):1–17.

Min Yang, Qiang Qu, Kai Lei, Jia Zhu, Zhou Zhao, Xiaojun Chen, and Joshua Z. Huang. 2018. *Investigating Deep Reinforcement Learning Techniques in Personalized Dialogue Generation*, pages 630–638.

Haolan Zhan, Sameen Maruf, Lizhen Qu, Yufei Wang, Ingrid Zukerman, and Gholamreza Haffari. 2023. Turning flowchart into dialog: Augmenting flowchart-grounded troubleshooting dialogs via synthetic data generation. In *Proceedings of the 21st Annual Workshop of the Australasian Language Technology Association*, pages 88–99, Melbourne, Australia. Association for Computational Linguistics.

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

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR '18, page 1097–1100, New York, NY, USA. Association for Computing Machinery.

A. Reinforcement Learning Agent Training Parameters

Parameter	Value
Optimizer	Adam
Learning Rate	$1e^{-4}$
λ	0.1
Maximum Training Dialog Turns	$2M$
Max. Gradient Norm	1.0
Batch Size	256
γ	0.99
Exploration fraction of Training Turns	0.99
Exploration Scheme	ϵ-greedy
ϵ start	0.6
ϵ end	0.0
Training frequency (w.r.t. dialog turns)	3
Training start (w.r.t. dialog turns)	1280
DDQN Target Network update frequency (w.r.t. training steps)	15
Q-Value clipping	10.0
Munchausen τ	0.03
Munchausen α	0.9
Munchausen Clipping	-1
Evaluation frequency (w.r.t. dialog turns)	10000
Evaluation dialogs	500

Table 8: Hyperparameters for training the Reinforcement Learning agents.

B. User Study

B.1. Data Agreement

Before beginning the experiment, users were provided with a data agreement. Although we did not collect any personally identifying data, we wanted to make sure that users were aware of what they would be asked to do, the purpose of the research, what data we would collect and how the data would be processed.

Impressum

Data Collection Policy

Please consider this information carefully before deciding whether to accept this task.

PURPOSE OF RESEARCH: To understand what expectations people have for task-oriented, text-based conversational agents and how these affect their interaction with such systems.

WHAT YOU WILL DO: You will be assigned to interact with one of three dialog systems. You will pretend that you are going on a business trip and interact with the assigned dialog system to find out answers to three different questions about the company's business travel regulations. Not all dialog systems will be able to deliver a good answer, if after trying, you cannot find an answer, you are free to move on to the next goal.

TIME REQUIRED: Participation will take approximately 15-20 minutes.

RISKS: There are no anticipated risks associated with participating in this study. The effects of participating should be comparable to those you would experience from viewing a computer monitor for 15-20 minutes and using a mouse and keyboard.

LIMITATIONS: This task is suitable for all people who can read from and input text into a computer.

CONFIDENTIALITY: Your participation in this study will remain confidential. Your responses will be assigned a code number. You will be asked to provide your Prolific ID, but this **will not be stored**, but rather converted to an anonymous hashed ID. You will be asked to provide your age and gender and previous experience with chatbots/business travel. Throughout the experiment, we may collect data such as your textual input, and your feedback in form of a questionnaire. The records of this study will be kept private. In any sort of report we make public we will not include any information that will make it possible to identify you. Research records will be kept in a locked file; only the researchers will have access to the records.

PARTICIPATION AND WITHDRAWAL: Your participation in this study is voluntarily, and you may withdraw at any time.

DATA REGULATION: Your data will be processed for the following purposes:
- Analysis of the respondents' evaluations of the dialog and their experience
- Analysis of potential influencing factors for individual behavior of the participants in the interaction with the dialog system
- Scientific publication based on the results of the above analyses

Your data will be processed on the basis of Article 6 paragraph 1 subparagraph 1 letter a GDPR. No personally identifying data will be collected. You are entitled to the following rights (for details see here)
- You have the right to receive information about the data stored about your person.
- Should incorrect personal data be processed, you have the right to correct it.
- Under certain conditions, you can demand the deletion or restriction of the processing as well as object to the processing.
- In general, you have a right to data transferability.
- Furthermore, you have the right of appeal to the Baden-Württemberg State Commissioner for Data Protection.

You can revoke your consent for the future at any time. The legality of the data processing carried out on the basis of the consent until revocation is not affected by this.

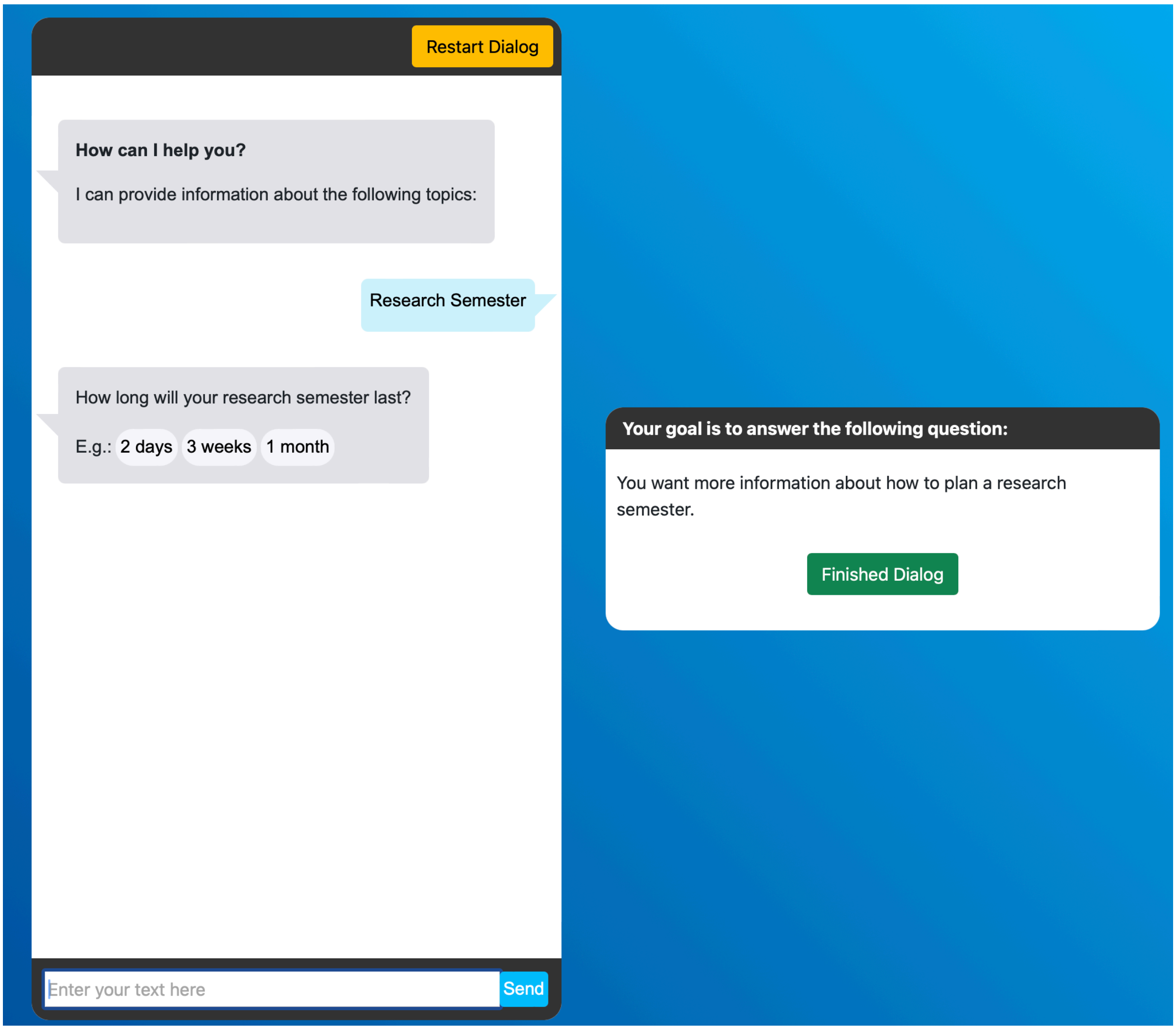
COMPENSATION: Upon completion of this task, you will receive a link to verify your completion with Prolific.

CONTACT: This study is conducted by researchers at the University of Stuttgart. If you have any questions or concerns about this study, please contact X at removed@for_anonymity.com

I agree

B.2. Study Instructions

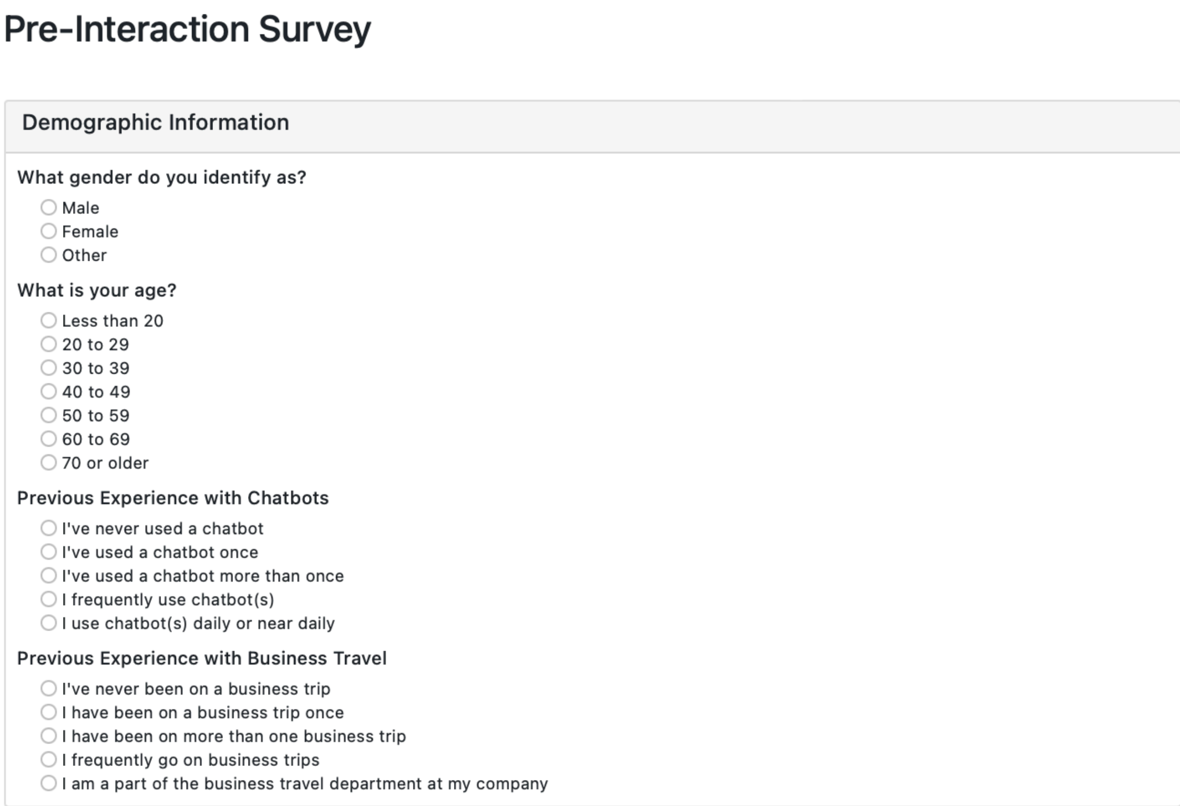
During the interaction, users were provided with the following interface, on the right side they had an information goal for which they should find an answer. On the left side, they had a window with their conversation with the chatbot. Once they felt they had found an answer to their question, they could click on the button underneath the goal to move on to the next dialog.



B.3. Interaction Surveys

B.3.1. Pre-Interaction Survey

The survey given to users before the interaction can be seen below. Here they were asked general questions about their demographics, previous experience with the domain and chatbots.



Pre-Interaction Survey

Demographic Information

What gender do you identify as?

- Male
- Female
- Other

What is your age?

- Less than 20
- 20 to 29
- 30 to 39
- 40 to 49
- 50 to 59
- 60 to 69
- 70 or older

Previous Experience with Chatbots

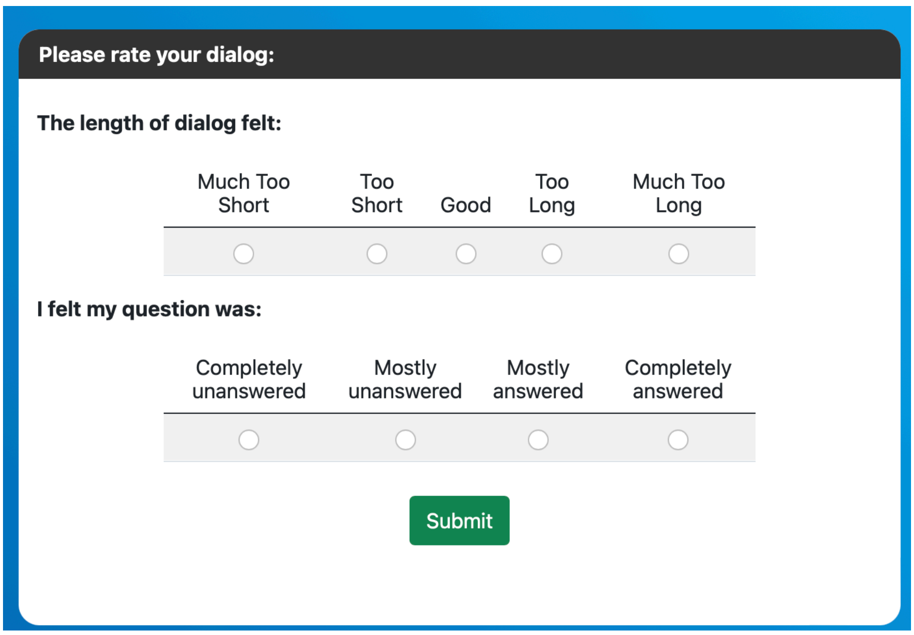
- I've never used a chatbot
- I've used a chatbot once
- I've used a chatbot more than once
- I frequently use chatbot(s)
- I use chatbot(s) daily or near daily

Previous Experience with Business Travel

- I've never been on a business trip
- I have been on a business trip once
- I have been on more than one business trip
- I frequently go on business trips
- I am a part of the business travel department at my company

B.3.2. Post-Dialog Survey

After each interaction, users were asked to rate their perception of the dialog length on a five-point Likert scale and their perception of how well their question was answered on a four-point Likert scale.



Please rate your dialog:

The length of dialog felt:

Much Too Short	Too Short	Good	Too Long	Much Too Long
○	○	○	○	○

I felt my question was:

Completely unanswered	Mostly unanswered	Mostly answered	Completely answered
○	○	○	○

Submit

B.3.3. Post-Interaction survey

The survey given to users after interacting with their assigned style of chatbot can be seen below. Users were asked to fill out a usability questionnaire (Finstad, 2010) and the trust and reliability subscales from the trust in automation questionnaire (Körber, 2018).

User Experience

Please mark how much you agree with the following statements:

	Strongly Disagree	Disagree	Neither Agree Nor Disagree	Agree	Strongly Agree
The chatbot was capable of interpreting situations correctly	○	○	○	○	○
The chatbot works reliably	○	○	○	○	○
A malfunction of the chatbot is likely	○	○	○	○	○
The chatbot is capable of handling complex tasks	○	○	○	○	○
The chatbot might make sporadic errors	○	○	○	○	○
I am confident about the chatbot's abilities	○	○	○	○	○
I trust the chatbot	○	○	○	○	○
I can rely on the chatbot	○	○	○	○	○
This chatbot's capabilities met my requirements	○	○	○	○	○
Using this chatbot is a frustrating experience	○	○	○	○	○
This chatbot is easy to use	○	○	○	○	○
I have to spend to long correcting things with this chatbot	○	○	○	○	○