Making Task-Oriented Dialogue Datasets More Natural by Synthetically Generating Indirect User Requests

Amogh Mannekote[*]
University of Florida
amogh.mannekote@ufl.edu

Jinseok Nam
Amazon
jinseo@amazon.com

Ziming Li
Amazon
ziminli@amazon.com

Kristy Elizabeth Boyer
University of Florida
keboyer@ufl.edu

Bonnie J. Dorr
University of Florida
bonniejdorr@ufl.edu

Abstract

Indirect User Requests (IURs), such as "It's cold in here" instead of "Could you please increase the temperature?" are common in human-human task-oriented dialogue and require world knowledge and pragmatic reasoning from the listener. While large language models (LLMs) can handle these requests effectively, smaller models deployed on virtual assistants often struggle due to resource constraints. Moreover, existing task-oriented dialogue benchmarks lack sufficient examples of complex discourse phenomena such as indirectness. To address this, we propose a set of linguistic criteria along with an LLM-based pipeline for generating realistic IURs to test Natural Language Understanding (NLU) and Dialogue State Tracking (DST) models before deployment in a new domain. We also release INDIRECTREQUESTS, a dataset of IURs based on the Schema Guided Dialog (SGD) corpus, as a comparative testbed for evaluating the performance of smaller models in handling indirect requests.

Utterance	Slot Value
Do you know if there are places that do the whole wine pairing thing with the meal around here?	**serves_alcohol** {**True**, False}
I usually watch Netflix on this device, can we play the song there?	**playback_device** {**TV**, kitchen speaker, bedroom speaker}

Figure 1: Two settings are illustrated for IURs: restaurant-reservation and home-automation.

1 Introduction

Non-literal, indirect utterances are common in human-human task-oriented dialogue and require pragmatic understanding and world knowledge for successful interpretation (e.g., *"It's cold in here"* instead of *"Could you please increase the temperature?"*) (Briggs and Scheutz, 2017). This phenomenon is a key area of interest in discourse pragmatics (Blum-Kulka and Hamo, 2011; Schegloff, 1999), supported by theoretical frameworks such as Grice's maxims (Grice, 1975) and RST (Mann and Thompson, 1988). Figure 1 illustrates two instances of Indirect User Requests (IURs).

Despite the prevalence of indirect utterances in everyday discourse and the human-level Natural Language Understanding (NLU) performance demonstrated by state-of-the-art large language models (LLMs) like GPT-4 (Achiam et al., 2023), current virtual assistants struggle to handle such utterances seamlessly (Mavrina et al., 2022). This can be attributed, in part, to the high computational cost associated with using state-of-the-art, large models for inference (Samsi et al., 2023; Sardana and Frankle, 2023). A common workaround is to employ smaller, cost-effective, task-specific models (Hsieh et al., 2023). However, this approach often compromises the generalizability and robustness offered by larger models.

Over the years, several benchmark datasets for task-oriented dialogue, such as MultiWOZ (Budzianowski et al., 2018), Schema Guided Dialog (SGD) (Rastogi et al., 2020), and FRAMES (Asri et al., 2017), have been curated by the dialogue systems community. However, these datasets have two key limitations that hinder their effectiveness in training smaller NLU models. First, their static nature and limited domain coverage make it difficult to evaluate NLU or Dialogue State Tracking (DST) models in new domains. Second, the controlled laboratory settings in which these datasets are crowdsourced lead to a distributional mismatch between the benchmark datasets and "in-the-wild" utterances (Zarcone et al., 2021).

[*]Work done during an internship at Amazon

Proceedings of the 31st International Conference on Computational Linguistics, pages 10449–10459
January 19–24, 2025.

2 Schema-Guided Dialogue

To bridge this distributional gap, we present an LLM-based data generation pipeline to scalably generate IURs for a new task-oriented dialogue domain. Our work makes the following contributions:

1. We develop a set of linguistic criteria to formalize the concept of what constitutes an indirect user request in a task-oriented dialogue setting.

2. We develop a pipeline to collect gold-labelled IURs, using an LLM to generate a noisy, seed IUR dataset, followed by crowd-sourced filtering and correction to increase quality.

3. We publicly release INDIRECTREQUESTS, a dataset of IURs collected through the process above, using the schemas from the SGD dataset. We aim for it to serve as a testbed for both researchers and practitioners interested in evaluating model robustness.

4. To circumvent the need for collecting expensive human labels for a new domain, we report results over various "proxy" models for *automatically* evaluating the quality of IURs according to our linguistic criteria.

5. Finally, we empirically demonstrate the increased difficulty of the IURs by showing that the performance of both a T5-based (Roberts et al., 2019) DST model as well as Llama 2 based DST models suffer significant degradation when applied on INDIRECTREQUESTS as compared to their counterparts from SGD.

Before outlining the linguistic criteria, we first describe the paradigm of "schema-guided dialogue" since it serves as the basis for the task formulation.

A long-standing goal in task-oriented dialogue research has been zero-shot transfer of critical modules such as the NLU and DST to previously unseen domains and backend APIs (Mehri et al., 2022). To achieve this goal, we need a way to represent new domains and APIs in a format that can be fed to a machine learning model. In addition, it helps if the representation is made as succinct to achieve both conceptual simplicity and human readability (Mannekote et al., 2023). A "dialogue schema" is any structured format that describes the domain that a dialogue system will operate in.

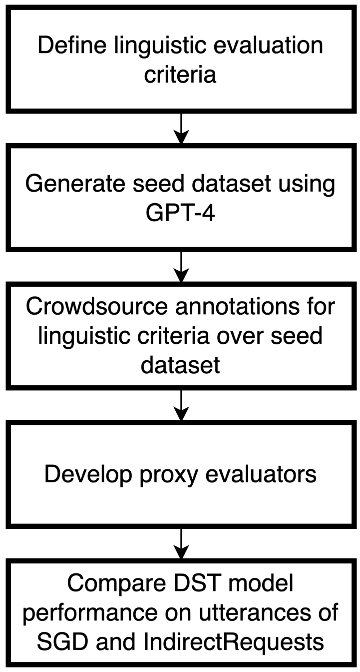

Figure 2: The five-stage IUR generation pipeline.

To facilitate shared tasks, Rastogi et al. (2020) formally introduce the paradigm of "schema-guided dialogue" alongside a benchmark corpus: the SGD dataset. Their schemas (shown in Figure 3) factor each task-oriented dialogue domain into its constituent *intents* and *slots*.

Consider two intents: `RentMovie` and `BuyTickets`. To satisfy each intent, the user needs to fill a set of slots. Slots can be considered analogous to query fields for an API call. For example, to fulfill the `BuyTickets` intent, the schema can demand that the `NumPeople`, `MovieName`, and `Date` slots be filled. A crucial aspect of SGD's schemas is their use of one-line natural language descriptions to describe the domain, intents, and slots. This design allows language models to make effective use of the schemas.

3 Linguistic Criteria

Effective NLU in task-oriented dialogue systems requires models to process and interpret user utterances in ways that are coherent, precise, and contextually appropriate. IURs depend on pragmatic reasoning and shared knowledge, as outlined in Grice's Maxims of *Relevance* and *Manner* (Grice, 1975) and Clark's theory of *common ground* (Clark, 1991). Building on these principles, we evaluate indirect user utterances across three dimensions: APPROPRIATENESS, UNAMBIGUITY, and WORLD-UNDERSTANDING, ensuring they are realistic and interpretable for robust NLU benchmarking.

We propose evaluating indirectness using three linguistic criteria: APPROPRIATENESS, UNAMBIGUITY, and WORLD-UNDERSTANDING. For each criterion, Table 1 shows examples of utterances that fall on the extreme ends of the rating scales. Note that each of the three labels carries a more precise meaning as compared to their freer usage in everyday language.

APPROPRIATENESS. The APPROPRIATENESS criterion seeks to ensure that an IUR does not sound out of place in the real-world context it is being uttered in. For instance, the utterance *"I'd like to order a sandwich"* would be completely irrelevant in a setting where the user is trying to book bus tickets. In contrast, the utterance *"I want to go somewhere"* would be relevant.

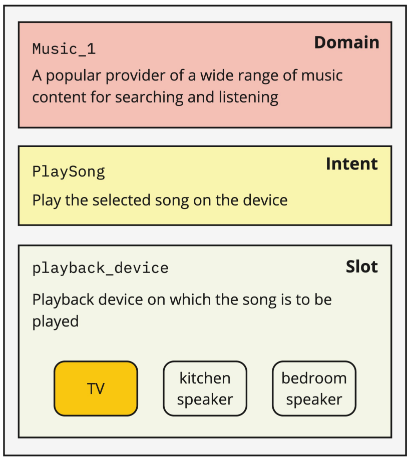

Figure 3: We illustrate a dialogue schema in the music service domain, with an intent to play music and a slot for selecting a playback device (e.g., TV, kitchen speaker, bedroom speaker). We generate an indirect utterance based on a specified slot value, such as 'TV.'

UNAMBIGUITY. The UNAMBIGUITY criterion is designed to ensure that a generated IUR entails the target slot value, not any of the remaining candidate slot values. For instance, a flight-booking scenario includes a "seating class" slot with values such as "Economy," "Premium Economy," "Business," and "First Class." Thus, the utterance *"I'm looking to book a luxurious seat on the flight"* is ambiguous, since the user could arguably be referring to any of these values.

WORLD-UNDERSTANDING. The WORLD-UNDERSTANDING criterion is intended to be a measure of the degree of world understanding required by the listener to draw the connection between an IUR and the user's intended target slot value. For example, when filling the *destination-country* slot in a trip-booking scenario, the utterance *"I'm looking to book a ticket to an African country"* can refer to values such as "Nigeria" or "Egypt" but not "India."

4 The INDIRECTREQUESTS Dataset

The goal of IUR generation is to take a domain, a domain schema, and a target slot value as inputs and output an IUR. The IUR, on its part, is expected to adhere to certain "linguistic criteria" to be valid.

Given a set of linguistic criteria for evaluating the quality of text samples, there are two broad approaches to crowdsource a dataset: (1) present real-world scenarios to crowdworkers and ask them to compose corresponding IURs in an open-ended manner, or (2) provide pre-generated IURs and ask crowdworkers to rate the quality of each IUR on a numerical scale reflecting the desired linguistic criteria. While the first approach demands crowdworkers to apply the provided linguistic framework, exhibit creativity, and possess proficient writing skills, rendering it expensive, the second approach involves the simpler task of evaluating existing utterances. Therefore, we generate a large number of (potentially noisy) IURs using a combination of GPT-3.5 (Brown et al., 2020) and GPT-4 models from OpenAI, and then ask crowdworkers to rate their quality based on our linguistic criteria.

4.1 Generating the Seed Dataset

In order to prompt an LLM for a task, we need a prompting strategy (operationalized using what is commonly referred to as a "prompt template"). While prompt engineering is an open-ended process, we follow guiding principles such as making instructions specific and detailed, including high-quality in-context examples, and exploiting strategies like Chain-of-Thought (CoT) (Wei et al., 2022) to improve output quality. We use CoT prompting (Wei et al., 2022) to generate IURs, as it has been shown to improve performance on NLP tasks involving reasoning, such as ours. This technique breaks down a problem into intermediate steps. For our task, we first generated a set of "interesting facts" about the target slot value in the given situation context, and then generated the final IURs conditioned on those facts. Therefore, this strategy was employed to scale up and generate a comprehensive seed dataset consisting of 453 IURs.

Linguistic Criterion	High-Scoring Utterance	Low-Scoring Utterance	Justification
APPROPRIATENESS	*I'm looking for tickets that I can exchange or refund in case of a change in plan.*	*I'd like to order a sandwich.*	The low-scoring example is nonsensical in the context of buying a bus ticket.
UNAMBIGUITY	*I'm looking for tickets that I can exchange or refund in case of a change in plan.*	*I'm looking for tickets that give me additional benefits.*	The term "additional benefits" is ambiguous as it can refer to either *Flexible* or *Economy Extra*.
WORLD-UNDERSTANDING	*Do you know of any Michelin star restaurants in the area that offer a unique dining experience?*	*I'm looking to treat myself to a luxurious meal with the highest quality ingredients.*	"Michelin star" demonstrates more in-depth world knowledge as opposed to "luxurious meal."

Table 1: Criteria to Evaluate IURs are provided with two accompanying example utterances: one that is high-scoring on that criterion, and another that is low-scoring.

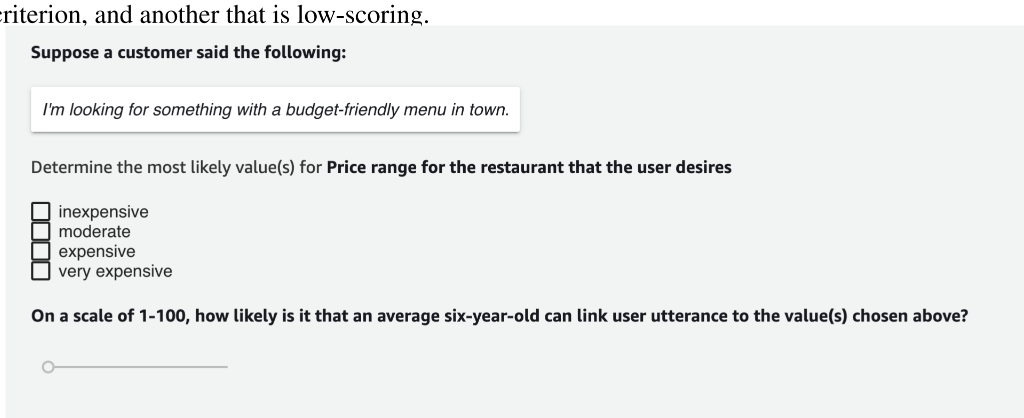

Figure 4: The M-Turk crowdsourcing interface for collecting human annotations over the seed dataset contains two form elements. The first assesses the UNAMBIGUITY in the generated utterance, ensuring that it entails only the target slot value. The second assesses the WORLD-UNDERSTANDING criterion, leveraging a slider to rate the likelihood that an average six-year-old could correctly infer the target slot value. The latter is an intuitive proxy to measure the complexity of world understanding required to interpret the utterance.

4.2 Crowdsourcing Human Labels

Manual inspection of the IURs in the seed dataset reveals considerable variation in quality, suggesting a need for refinement before utilizing them as gold-labeled data for evaluation. To address this, we set up a crowdsourcing pipeline using Amazon Mechanical Turk (M-Turk) to have crowdworkers rate the quality of the candidate IURs in accordance with our linguistic criteria.

There are two key considerations for developing the crowdsourcing interface: 1) to optimize annotator efficiency (reducing the time and effort required per evaluated sample) and 2) to maximize inter-annotator agreement. We observe that the variation in the unannotated seed dataset is predominantly along the criteria of UNAMBIGUITY and WORLD-UNDERSTANDING. Only a negligible number of instances were deemed irrelevant based on the APPROPRIATENESS criteria. Consequently, we streamline the interface to include two primary components, one each for evaluating UNAMBIGUITY and WORLD-UNDERSTANDING.

UNAMBIGUITY Annotation. To collect labels for the UNAMBIGUITY criterion, we instruct the annotators to select all the slot values (zero or more) that they think are entailed by the utterance using a multiple choice checkbox (the annotator can check one or more boxes). We design this form element as a binary yes/no question to avoid posing the question in a leading way. Multiple selections by an annotator imply the utterance fails to meet the UNAMBIGUITY criterion.

WORLD-UNDERSTANDING Annotation. For the WORLD-UNDERSTANDING criterion, we ask annotators to engage in a thought experiment where they adopt the perspective of a six-year-old child. This approach aims to assess whether a connection between the utterance and selected slot values would be discernible to a child of that age. We arrived at this unique framing after several iterations of refining the question. Initially, we asked annotators directly to rate the "complexity" involved in making the connection. However, we recognized that the concept of "complexity" is highly subjective and can vary significantly among individuals. To standardize the perception of complexity and reduce variability among annotators, we anchor our assessment to a child's level of understanding. This approach aims to provide a consistent benchmark, despite the diverse cognitive abilities typically present at that age range.

4.3 Dataset Splits

Based on the crowdsourced labels for both UNAMBIGUITY and WORLD-UNDERSTANDING, we curate the INDIRECTREQUESTS dataset and release it for public use.[1] In going from the "raw" crowdsourced samples to the dataset, we split the dataset and systematically create labels for each sample for both UNAMBIGUITY and WORLD-UNDERSTANDING criteria. While splitting INDIRECTREQUESTS into train, validation, and test sets, we split our samples based on same lines on which the services are split across the SGD dataset. This alignment with the SGD dataset splits is intended to aid future work that might need to compare our results with previous work reporting on SGD.

Train	Validation	Test
123	136	194

Table 2: Number of samples in each split of INDIRECTREQUESTS

5 Proxy Evaluation of Linguistic Criteria

We perform an automated, proxy evaluation of the IURs generations due to the impracticality of manually evaluating the large number of samples and models. In this section, we define the proxy evaluation task formulations and present baseline results using zero-shot and few-shot prompting strategies. We define two proxy evaluation tasks, corresponding to the UNAMBIGUITY and WORLD-UNDERSTANDING criteria, respectively.

UNAMBIGUITY. We frame proxy evaluation of UNAMBIGUITY as a multi-class classification problem with $N_i + 1$ classes, where N_i is the number of possible slot values for the given slot i. We add an extra class corresponding to the case where the ground truth (from the crowdsourcing step) is ambiguous. For model comparison, we report the accuracy over all samples in the test split.

WORLD-UNDERSTANDING. We define the proxy evaluation of WORLD-UNDERSTANDING as predicting the level of world knowledge required to infer the intended slot value from an utterance as a continuous value ranging from 1 to 10. This approach aligns with the methodology used in our crowdsourcing stage, where judgments about knowledge depth were made using a 1-100 scale slider. Performance is quantified by calculating the sum of squared errors between predicted and actual values (after normalizing both sets of values).

5.1 Proxy Evaluation Results

We split the proxy evaluation models into three categories: small language models (fewer than 1B parameters), proprietary large language models from OpenAI (gpt-3.5-turbo and gpt-4-0125-preview), and open-source Llama 2 language models (7B, 13B, and 70B) - both base and chat variants. Table 3 shows the performance of the proxy evaluators on the test split against the ground truth obtained through crowdsourcing.

Small LMs. For the small LM category, we employ BERT-based models in a zero-shot setup. For the UNAMBIGUITY criterion, we frame the evaluation as k Natural Language Inference (NLI) problems, where k is the number of possible slot values. Each problem considers the candidate IUR as the premise and a possible slot value as the hypothesis. We use a BERT-based NLI model[2] to obtain entailment scores and return the argmax score. If the maximum score is below 0.3, we deem the IUR ambiguous for that slot. For WORLD-UNDERSTANDING, we use ms-marco-MiniLM-L-6-v2[3], fine-tuned on MS MARCO for passage ranking. We concatenate the IUR with the knowledge context, score the sequence using the model, and assign a WORLD-UNDERSTANDING rating of 10 if the the score exceeds 0.5 and 0 otherwise.

Proprietary LLMs. For the proprietary LLMs from OpenAI, we use the models in a few-shot setup, providing a few examples of IURs labeled

[1] https://huggingface.co/datasets/msamogh/indirect-requests

[2] nli-deberta-v3-small

[3] https://huggingface.co/microsoft/ms-marco-MiniLM-L-6-v2

Criterion	Model								
	Small LM (<1B)	GPT (3-shot)		Llama 2 Base (3-shot)			Llama 2 Chat (3-shot)		
		GPT-3.5	GPT-4	7B	13B	70B	7B	13B	70B
UNAMBIGUITY (Accuracy)	0.35* (nli-deberta)	0.73+	0.84†	0.50	0.69‡	0.22	0.55	0.62	0.73+
WORLD-UNDERSTANDING (Pearson correlation)	0.22* (ms-marco)	0.15	0.34†	0.16	0.19‡	0.18	0.17	0.18	0.20+

Table 3: Evaluation results are computed from a single run with proxy evaluators against crowdworker annotations on the combined validation and test splits of INDIRECTREQUESTS, which contain a total of 330 samples. Performance symbols indicate the best-performing models within specific categories. * denotes the best performance in the zero-shot (small LM) category, † marks the best performance in the proprietary OpenAI LLM category, ‡ signifies the top performer among the Llama 2 Base models (Touvron et al., 2023), and + represents the best performing variant of the Llama 2 Chat models.

as either ambiguous or unambiguous (for UNAMBIGUITY), or knowledgeable or not knowledgeable (for WORLD-UNDERSTANDING). We then query the model with the test IUR and and take the model's output as the prediction.

Open-Source LLMs. For the open-source Llama 2 models (7B, 13B, and 70B), we use a similar few-shot setup as we did with the proprietary LLMs. We experiment with both the base and the chat model variants. Table 3, summarizes these results.

While achieving high inter-annotator agreement (IAA) for subjective measures like WORLD-UNDERSTANDING and UNAMBIGUITY is inherently challenging, as evidenced by prior work showing human annotators struggling to exceed 30% IAA for related subjective criteria in NLG tasks (Karpinska et al., 2021), we find that LLM-based proxy evaluation models, particularly GPT-3.5 and GPT-4, demonstrate considerable agreement with human raters for our task. Nonetheless, there remains scope for further boosting performance through additional prompt engineering and experimentation with adaptive strategies for selecting in-context examples. The prompts used for training both proprietary and open-source LLM proxy evaluator models are provided in Appendix B.

6 Automated IUR Generation

Under ideal conditions, we would use as small an LLM as possible to generate high-quality IURs. We report the quality of the generated IURs generated using smaller, open-source LLMs (Llama 2) in Figure 5. The prompt used to generate the IURs is given in Appendix C.

6.1 Indirection Strategies

Along with reporting quantitative metrics from our proxy evaluators, we also perform a bottom-up content analysis to gain a richer understanding of the specific "indirection strategies" that LLMs employ to transform the slot schema into IURs. During analysis, one of the authors excluded samples for which the IUR either very evidently does not entail the target slot value or the slot value is mentioned verbatim, violating the UNAMBIGUITY criterion.

We identify five main indirection strategies from our content analysis (see Table 4). **Simple Elaboration** performs a simple replacement of the slot value with a longer phrase meaning the same thing. Simple Elaborations do not leverage non-trivial world knowledge. **Justification** offers a real-world reason for choosing a particular slot value. A **Hyponym Swap** involves replacing the slot value with its hyponym (the replacement is a more specific instance or subtype of the original term). Similarly, a **Synonym Swap** replaces the slot value with a synonym. The final strategy, **Small Talk**, involves padding the utterance with information that is not strictly informational to the task. While this is not strictly an indirection strategy, it can serve to complement another indirection strategy by making it sounds more realistic.

7 Extrinsic Evaluation

While intrinsic, automated evaluations provide valuable insights, we further assess the practical implications of INDIRECTREQUESTS through extrinsic evaluation, measuring the performance degradation of a widely-adopted DST model on our dataset compared to its performance on the canonical SGD corpus. This approach aligns with established practices in the dialogue systems literature, where NLU model performance is extensively evaluated in isolation, as it critically impacts downstream dialogue policy and response generation.

Our objective is not to conduct an end-to-end

Indirection Strategy	Intent-Slot-Value	Sample IUR
Simple Elaboration	RentMovie (subtitles = None)	"I prefer watching films in their native language **without any language barriers**."
Justification	GetRide (shared_ride = True)	"I usually like sharing the ride with someone else **to reduce carbon footprint**..."
Hyponym Swap	SearchEvents (type = Music)	"Is there a festival happening around with **pop**, **country** or **hip-hop** artists performing?"
Synonym Swap	RentMovie (subtitles = Mandarin)	"I've got a bunch of friends coming over who are more comfortable with **Simplified Chinese**. Can you find me movies..."
Small Talk	FindApartment (pets_allowed = True)	"I'm looking for a place where my dog is allowed to come along. **He's so cute and he doesn't shed as much as you think!**"

Table 4: From the generated IURs, we identify five main indirection strategies (Simple Elaboration, Justification, Hyponym Swap, Synonym Swap, and Small Talk).

evaluation of dialogue systems, but to specifically evaluate NLU performance. By providing a relative comparison against the commonly referenced SGD corpus, we aim to highlight the increased parsing difficulty posed by INDIRECTREQUESTS utterances, rather than claiming they present challenges to state-of-the-art models, including LLM-based ones. This targeted evaluation allows us to isolate and characterizet unique aspects of our dataset, providing a more comprehensive understanding of NLU model capabilities and limitations.

Since the DST model we use is trained on context windows of length 3, the dialogue contexts in all samples are also set to the same length. Table 5 shows a comparison between the model performance over the original samples and the samples using the generated IURs based on a total of 330 samples, with statistically significant performance results for all models.

To fairly compare the results of any NLU model over SGD and INDIRECTREQUESTS during extrinsic evaluation, we only use a subset of SGD that satisfies the following conditions:

1. user request must be about a categorical slot
2. speaker of the latest utterance in the dialogue context must be the user and not the system
3. dialogue act of the latest utterance should be "inform" (as opposed to "request" utterances, which is out of scope for our work)
4. user utterance includes only a single slot-value pair (since our IUR generation method does not accommodate more than one slot-value pair per IUR).

Model	SGD	INDIRECTREQUESTS	p-value
T5-DST	0.51	0.13 ↓	$< 10^{-8}$
Llama 2 7B	0.79	0.61 ↓	10^{-8}
Llama 2 13B	0.94	0.84 ↓	10^{-5}
Llama 2 70B	0.96	0.69 ↓	$< 10^{-8}$

Table 5: Comparison of slot accuracies between the indirect utterances from SGD and INDIRECTREQUESTS with an efficient DST model based on T5 as well as Llama 2 models (7B, 13B, 70B). All models show significant performance drops on INDIRECTREQUESTS, highlighting its increased difficulty.

8 Related Work

Brittleness of DST Models. The initiative to develop the IUR generation task springs from a need to reduce the brittleness of smaller NLU and DST models. Cho et al. (2022) empirically demonstrate the brittleness of commonly-used, small LM-based DST models by showing that their performance degrades in the face of various types of perturbations involving linguistic variations, coreferences, named entity references, paraphrases, and speech disfluencies. More generally, Zarcone et al. (2021) critique the academic community's prevailing focus on incremental advancements on synthetic benchmarks for tasks such as DST, referred to as *"playing the SNIPS game,"* which often overlooks deeper issues regarding dataset realism.

Relationship of IUR Generation to Other NLP Tasks. IUR generation is similar to paraphrase generation (Zhou and Bhat, 2021) in that both tasks are form of semantically-preserving text transformations. In fact, IUR generation can be viewed as the task of generating a highly specific form of paraphrase (that adheres to our three linguistic criteria). It can also be viewed as the inverse of the NLI task, where the objective is to generate a premise

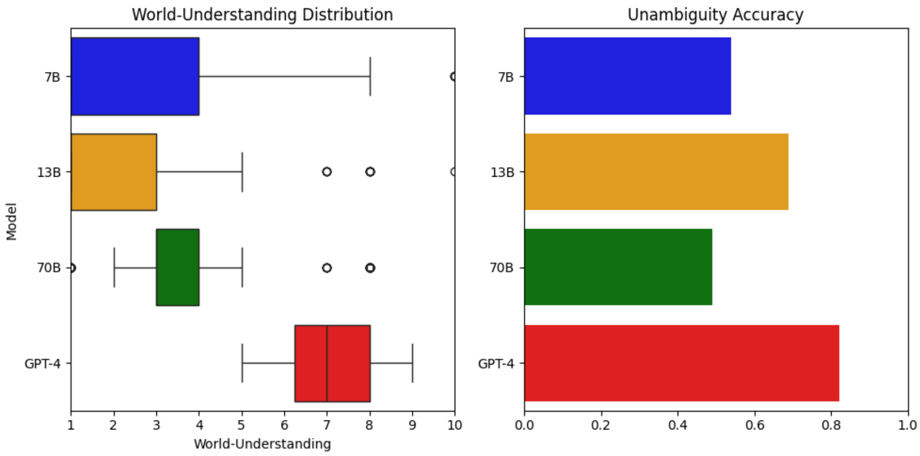

Figure 5: We report the qualities of the IURs generated using smaller, open-source Llama 2 models of three different sizes (7B, 13B, 70B). All the evaluation results are obtained using the best-performing GPT-4 proxy evaluation model (as described in Section 5).

entailing a given hypothesis, rather than inferring entailment from a premise-hypothesis pair, albeit in a different context from Shen et al. (2018). Most closely related to our work, Ge et al. (2022) propose linguistic criteria based on Gricean maxims (Grice, 1975) for the task of generating follow-up questions for interactive surveys. While both tasks prioritize relevance and coherence, they differ in their objectives: the former aims to elicit information from the user, while the latter focuses on clarity and unambiguity in conveying requests, often serving as the initial turn or an independent subdialogue thread.

Text Generation using Small LLMs. Our research examines the impact of model size on the quality of the generated IURs. Eldan and Li (2023) dispute the notion that smaller Language Models (LMs) inherently lack the capacity for intricate text generation tasks like storytelling. They attribute shortcomings to the prevalence of irrelevant information rather than model constraints. Using a targeted dataset of children's stories, they show that smaller LMs can produce narratives comparable to those by larger counterparts like GPT-3.5 and GPT-4. Our work is aligned with this broader spirit, aiming to match the output of a larger LLMs through fine-tuning a smaller model.

9 Limitations and Future Work

This work focuses on supervised fine-tuning of LLMs, but reinforcement learning offers potential for guiding models toward abstract concepts like *indirectness* (Kaufmann et al., 2023). LLMs may also encode biases due to their training data, which can be mitigated through calibration methods that provide confidence estimates alongside predictions.

As Bowman and Dahl (2021) highlights, real user data remains the gold standard for evaluating NLP tasks. While our work enhances the modeling of indirectness in task-oriented dialogue, broader evaluation paradigms are essential for diverse scenarios (Mannekote, 2023). The proposed linguistic criteria provide a foundation for future datasets, enabling expanded slot values to further challenge NLU models.

10 Conclusion

This study bridges the gap between benchmark corpora and real-world utterances in task-oriented dialogue by addressing indirectness. We introduce INDIRECTREQUESTS, a dataset of IURs generated using an LLM-based pipeline built on schemas from the SGD dataset of schema-guided dialogue. INDIRECTREQUESTS enables rigorous evaluation of NLU and DST models on realistic indirect requests. Experiments validate its challenging nature, while the pipeline offers an efficient method to create diverse evaluation datasets, enhancing the performance and usability of the virtual assistant. Moreover, this dataset captures nuanced language patterns that are often absent from traditional datasets. By incorporating indirect utterances, INDIRECTREQUESTS represents a significant advancement in the development of more robust and context-aware dialogue systems.

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*.

Layla El Asri, Hannes Schulz, Shikhar Sharma, Jeremie Zumer, Justin Harris, Emery Fine, Rahul Mehrotra, and Kaheer Suleman. 2017. Frames: a corpus for adding memory to goal-oriented dialogue systems. *arXiv preprint arXiv:1704.00057*.

Shoshana Blum-Kulka and Michal Hamo. 2011. Discourse pragmatics. *Discourse studies: A multidisciplinary introduction*, 2(1):143–164.

Samuel R. Bowman and George Dahl. 2021. What Will it Take to Fix Benchmarking in Natural Language Understanding? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4843–4855, Online. Association for Computational Linguistics.

Gordon Briggs and Matthias Scheutz. 2017. Strategies and mechanisms to enable dialogue agents to respond appropriately to indirect speech acts. In *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, pages 323–328. IEEE.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Inigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ–a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *arXiv preprint arXiv:1810.00278*.

Hyundong Cho, Chinnadhurai Sankar, Christopher Lin, Kaushik Ram Sadagopan, Shahin Shayandeh, Asli Celikyilmaz, Jonathan May, and Ahmad Beirami. 2022. Know Thy Strengths: Comprehensive Dialogue State Tracking Diagnostics. ArXiv:2112.08321 [cs].

HH Clark. 1991. Grounding in communication. *Perspectives on socially shared cognition/American Psychological Association*.

Ronen Eldan and Yuanzhi Li. 2023. TinyStories: How Small Can Language Models Be and Still Speak Coherent English? ArXiv:2305.07759 [cs].

Yubin Ge, Ziang Xiao, Jana Diesner, Heng Ji, Karrie Karahalios, and Hari Sundaram. 2022. What should i ask: A knowledge-driven approach for follow-up questions generation in conversational surveys. *arXiv preprint arXiv:2205.10977*.

Herbert P Grice. 1975. Logic and conversation. In *Speech acts*, pages 41–58. Brill.

Cheng-Yu Hsieh, Chun-Liang Li, Chih-Kuan Yeh, Hootan Nakhost, Yasuhisa Fujii, Alexander Ratner, Ranjay Krishna, Chen-Yu Lee, and Tomas Pfister. 2023. Distilling step-by-step! outperforming larger language models with less training data and smaller model sizes. *arXiv preprint arXiv:2305.02301*.

Marzena Karpinska, Nader Akoury, and Mohit Iyyer. 2021. The perils of using Mechanical Turk to evaluate open-ended text generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1265–1285, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Timo Kaufmann, Paul Weng, Viktor Bengs, and Eyke Hüllermeier. 2023. A survey of reinforcement learning from human feedback. ArXiv: 2312.14925 [cs.LG].

William C Mann and Sandra A Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text-interdisciplinary Journal for the Study of Discourse*, 8(3):243–281.

Amogh Mannekote. 2023. Towards a neural era in dialogue management for collaboration: A literature survey. *ArXiv*, abs/2307.09021.

Amogh Mannekote, Mehmet Celepkolu, Joseph B. Wiggins, and Kristy Elizabeth Boyer. 2023. Exploring usability issues in instruction-based and schema-based authoring of task-oriented dialogue agents. *Proceedings of the 5th International Conference on Conversational User Interfaces*.

Lina Mavrina, Jessica Szczuka, Clara Strathmann, Lisa Michelle Bohnenkamp, Nicole Krämer, and Stefan Kopp. 2022. "Alexa, You're Really Stupid": A Longitudinal Field Study on Communication Breakdowns Between Family Members and a Voice Assistant. *Frontiers in Computer Science*, 4.

Shikib Mehri, Yasemin Altun, and Maxine Eskenazi. 2022. Lad: Language models as data for zero-shot dialog. *arXiv preprint arXiv:2207.14393*.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8689–8696. Issue: 05.

Adam Roberts, Colin Raffel, Katherine Lee, Michael Matena, Noam Shazeer, Peter J Liu, Sharan Narang, Wei Li, and Yanqi Zhou. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *Google, Tech. Rep.*

Siddharth Samsi, Dan Zhao, Joseph McDonald, Baolin Li, Adam Michaleas, Michael Jones, William Bergeron, Jeremy Kepner, Devesh Tiwari, and Vijay Gadepally. 2023. From words to watts: Benchmarking the energy costs of large language model inference. In *2023 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–9. IEEE.

Nikhil Sardana and Jonathan Frankle. 2023. Beyond chinchilla-optimal: Accounting for inference in language model scaling laws. *arXiv preprint arXiv:2401.00448*.

Emanuel A Schegloff. 1999. Discourse, pragmatics, conversation, analysis. *Discourse studies*, 1(4):405–435.

Yikang Shen, Shawn Tan, Chin-Wei Huang, and Aaron Courville. 2018. Generating contradictory, neutral, and entailing sentences. *arXiv preprint arXiv:1803.02710*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Alessandra Zarcone, Jens Lehmann, and Emanuël AP Habets. 2021. Small data in nlu: Proposals towards a data-centric approach. In *35th Conference on Neural Information Processing Systems (NeurIPS 2021)*.

Jianing Zhou and Suma Bhat. 2021. Paraphrase generation: A survey of the state of the art. In *Proceedings of the 2021 conference on empirical methods in natural language processing*, pages 5075–5086.

A Instructions shown to Human Annotators

For each task (sample), the annotators were required to fill in a form with two input fields. We provided examples along with brief instructions on how to fill in these fields (see Figure 4) as shown below.

To get a feel for the task, please go through these examples.

In all the examples below, the customer is trying to search for restaurants and indicating their preference for "Italian cuisine."

1. **Check all entailing slot values:** *For the first question, you will need to check all the values that can be implied by the customer's utterance. This could mean selecting zero, one, or more checkboxes.* [examples]

2. **Use the slider to indicate the difficulty of the utterance.** [examples]

B Prompts for Proxy Evaluators

Below, we list the LLM prompts used for proxy evaluation of UNAMBIGUITY and WORLD-UNDERSTANDING criteria.

B.1 UNAMBIGUITY

```
You are an expert at
    ↪ evaluating which slot
    ↪ value(s) could be
    ↪ implied by an utterance
    ↪ among a set of
    ↪ candidate values in a
    ↪ task-oriented dialogue.
    ↪ If no values can be
    ↪ eliminated, list all
    ↪ possible values
    ↪ separated by commas.
Examples:
Situation: User wants to make
    ↪ a trip
Slot: Destination country
Possible Values: India,
    ↪ Namibia, Nigeria
Utterance: I'm looking to
    ↪ book a ticket to an
    ↪ African country
Slot Values Implied: Namibia,
    ↪ Nigeria

<more in-context examples>
```

B.2 WORLD-UNDERSTANDING

```
On a scale of 1-10, how
    ↪ likely is it that an
    ↪ average six-year-old
    ↪ would be able to link
    ↪ the user utterance to
    ↪ the target slot value?
Examples:
Situation: User wants to find
    ↪ concerts and games
    ↪ happening in your area
Slot: Destination country
Possible Values: India,
    ↪ Namibia, Nigeria
Utterance: I'm looking to
    ↪ book a ticket to an
    ↪ African country
World Knowledge Level: 10
```

```
<more in-context examples>
```

C Prompt for Generating IURs

Below is the prompt used to generate IURs.

```
Generate a customer utterance
   ↪ containing an indirect and
   ↪ unique reason for wanting
   ↪ to choose a target slot
   ↪ value. Make sure that 1)
   ↪ the utterance entails ONLY
   ↪ the target slot value and
   ↪ that it DOES NOT mention
   ↪ the target slot value.

Situation: User wants to
   ↪ transfer money from one
   ↪ bank account to another
   ↪ user's account
Slot Description: The account
   ↪ type of the recipient whom
   ↪ the user is transfering
   ↪ money to
Possible Slot Values: checking,
   ↪ savings
Target Slot Value: checking
Do Not Mention: checking
Indirect User Request Keywords
   ↪ In: I need to transfer
   ↪ some money to my friend's
   ↪ account. He usually uses
   ↪ it for his direct deposits.

Situation: User wants to find a
   ↪ restaurant of a particular
   ↪ cuisine in a city
Slot Description: Price range
   ↪ for the restaurant
Possible Slot Values:
   ↪ inexpensive, moderate,
   ↪ expensive
Target Slot Value: moderate
Do Not Mention Keywords In:
   ↪ moderate
Indirect User Request: Looking
   ↪ to have a decent meal
   ↪ without burning a hole in
   ↪ my pocket

Now, generate ONE indirect user
   ↪ request for this input
   ↪ based on the above
   ↪ examples.
Situation: {situation}
Slot Description:
   ↪ {slot_description}
Possible Slot Values:
   ↪ {possible_slot_values}
Target Slot Value:
   ↪ {target_slot_value}
Do Not Mention Keywords In:
   ↪ {target_slot_value}
```

D Prompt for Extrinsic Evaluation

```
For the given task-oriented
   ↪ dialogue, predict the most
   ↪ likely target value the
   ↪ user is requesting for the
   ↪ given slot. Output without
   ↪ quotes. If it's ambiguous,
   ↪ output "<ambiguous>" Just
   ↪ output the target slot
   ↪ value from the possible
   ↪ slot values. Nothing else.
Previous System Utterance:
   ↪ {prev_sys_utterance}
User request: {utterance}
Slot Description:
   ↪ {slot_description}
Possible Slot Values:
   ↪ {possible_slot_values}
Target Slot Value:
```

E Generation Parameters

OpenAI Models. We use the default settings from the OpenAI for our experiments with GPT-3.5 and GPT-4 models.

Llama 2 Models. For all generation experiments with Llama 2, we use the following parameters.

Top-k: 50

Top-p: 0.9

Temperature: 0.5

Max New Tokens: 128

Min New Tokens: -1

Stop Sequences: `\n`