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

Anonymous ACL submission

Abstract

Existing benchmark corpora of task-oriented dialogue are collected either using a "machines talking to machines" approach or by giving template-based goal descriptions to crowdworkers. These methods, however, often produce utterances that are markedly different from natural human conversations in which people often convey their preferences in indirect ways, such as through small talk. We term such utterances as Indirect User Requests (IURs). Understanding such IURs demands considerable world knowledge and reasoning capabilities on the listener's part. Our study introduces a large language model (LLM)-based pipeline to automatically generate realistic, high-quality IURs for a given domain, with the ultimate goal of supporting research in Natural Language Understanding (NLU) and Dialogue State Tracking (DST) for task-oriented dialogue systems. Our findings show that while large LLMs such as GPT-3.5 and GPT-4 generate high-quality IURs, achieving similar quality with smaller models is more challenging. We release INDIRECTREQUESTS, a dataset of IURs that advances beyond the initial Schema Guided Dialog (SGD) dataset in that it provides a challenging testbed for testing the "in the wild" performance of NLU and DST models.

1 Introduction

Task-oriented dialogue assistants (Balaraman et al., 2021) help people carry out tasks such as making hotel reservations, setting alarms, looking up train schedules, and so on through natural language conversations (Budzianowski et al., 2018; Mosig et al., 2020; Byrne et al., 2019; Asri et al., 2017). One of the most challenging aspects of developing a dialogue assistant is developing the natural language understanding model (Mehri et al., 2020). With the proliferation of powerful LLMs (Brown et al., 2020), great strides have been made in model performance on a handful of benchmark datasets (Budzianowski et al., 2018; Rastogi et al., 2020; Asri et al., 2017) that are widely regarded as proxies for the general task. However, despite these advances, many models suffer from errors when presented with utterances that differ in particular ways from those present in their training datasets (Cho et al., 2021, 2022). As a result, users end up feeling frustrated when using talking to these virtual assistants freely as they would with other humans (Mavrina et al., 2022).

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.

There are several failure modes of NLU and DST models, one of them being the lack of model capability to understand indirect requests that do not directly mention the target slot value as expected by the system (Cohen, 2019). For example, while reserving a hotel room, rather than saying the presumably more natural utterance, *"it's gonna be me, my wife, and our twins"*,[1] a user might instead resort to more *direct* terms (for example, *"I want to book a room for four people"*) to ensure that the intent of the utterance is understood by the virtual assistant on the first attempt. Figure 1 shows two notional instances of IURs in the context of a restaurant reservation and an intelligent home-assistant dialogue respectively.

From a machine learning standpoint, the challenge DST models encounter in understanding

[1]This is presumably a natural thing to say during a friendly chat with a human receptionist.

IURs stems from the lack of labeled examples in mainstream benchmark datasets used for developing task-oriented dialogue agents (Cho et al., 2021). Furthermore, the main reason for this discrepancy in distribution between benchmark datasets and "in-the-wild" utterances can be attributed to the controlled environment of laboratory settings where datasets are crowdsourced (Zarcone et al., 2021).

Existing benchmark datasets of task-oriented dialogue such as MultiWOZ (Budzianowski et al., 2018), Schema Guided Dialog (SGD) (Rastogi et al., 2020), and FRAMES (Asri et al., 2017) all suffer from this distributional mismatch. For example, we found that only around 500 out of over 10,000 user utterances in the SGD dataset do not contain an explicit mention of the target slot value.

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 contributes the following:

1. We suggest linguistic criteria to formalize the concept of what constitutes an *indirect* IUR. Specifically, we develop a set of linguistic criteria to systematically evaluate questions such as *"What counts as an indirect user request?"* and *"How indirectly is this user request phrased with respect to a given domain schema?"* in task-oriented dialogue contexts.

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 schefmas 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 a state-of-the-art DST model significantly degrades when applied on INDIRECTREQUESTS utterances as compared to their counterparts from SGD.

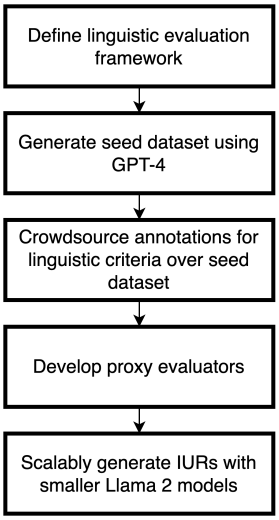

Figure 2: The five-stage IUR generation pipeline.

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

2 Schema-Guided Dialogue

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. A "dialogue schema" is any structured format that performs this role of describing a domain that a dialogue system will operate in.

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 a `Movie` domain consisting of 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.

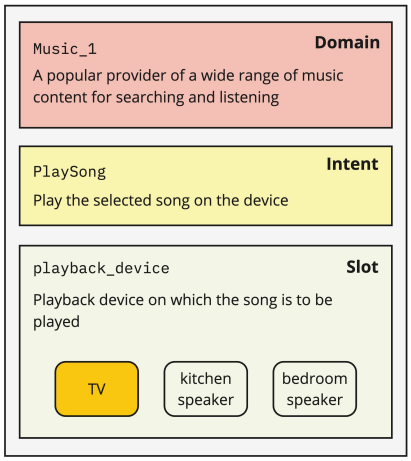

Figure 3: We illustrate a dialogue schema in the music service domain, with an intent to play music and a slot for selecting a playback device (e.g., TV, kitchen speaker, bedroom speaker). Our approach generates an indirect utterance based on a specified slot value, such as 'TV.'

3 Linguistic Criteria

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.

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

Given a linguistic framework for evaluating the quality of text samples (such as ours), there are two broad approaches to crowdsource a dataset.

1. present real-world scenarios to the crowdworkers and ask them to compose corresponding IURs in an open-ended way, or

2. provide crowdworkers with pre-generated IURs and ask them to *rate* the quality of each IUR on a numerical scale that reflects our desired linguistic criteria.

The first approach, where crowdworkers are given our linguistic framework and asked to come up with IURs based on it, demands creativity and proficient writing skills, making it expensive. In contrast, the second approach involves workers evaluating existing utterances, which is simpler. Therefore, we generate a large number of (potentially noisy) IURs using a combination of GPT-3.5[2] (Brown et al., 2020) and GPT-4[3] models from OpenAI, and then ask crowdworkers to rate their quality based on our linguistic criteria.

4.1 Prompting Strategies for Generating 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 experiment with three prompting strategies for generating our seed dataset:

1. **Zero-Shot Prompt (Instruction-Only):** Prompting the LLM with only a natural

[2]For the rest of this paper, when we say GPT-3.5, we mean `gpt-3.5-turbo`.

[3]Similarly, when we say GPT-4, we mean `gpt-4-0125-preview`.

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, so I'd like to find a restaurant like that*	"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.

language instruction containing a description of the linguistic framework.

2. **Few-Shot Prompt (Instruction + In-Context Examples):** In addition to 1 above, we experiment with adding a few "in-context" examples that correspond to human-labelled gold-standard samples.

3. **Few-Shot Prompt with CoT:** Using CoT prompting (Wei et al., 2022), a technique that 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.

Two of the authors of this paper sampled a handful of IURs generated from all three prompting strategies and determined that the **Few-Shot Prompt with CoT** strategy resulted in IURs that were the most realistic looking. Hence, we scale up this strategy to generate a seed dataset of 453 IURs.

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.

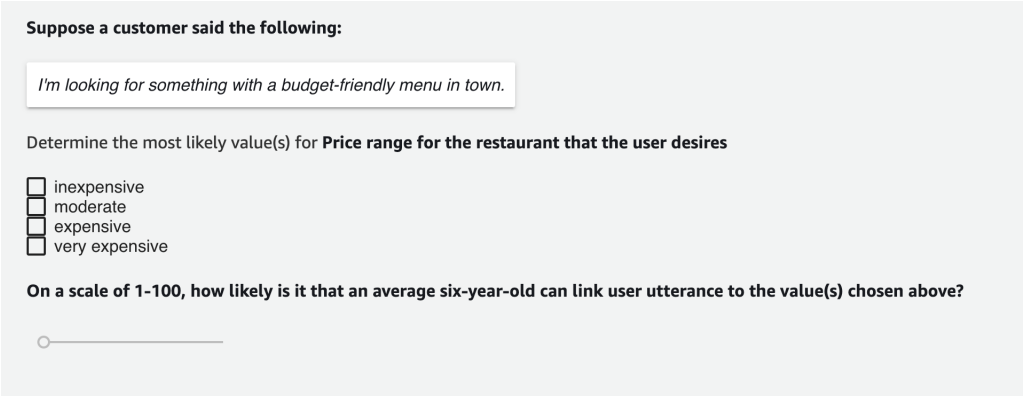

Figure 4: The M-Turk crowdsourcing interface for collecting human annotations over the seed dataset contains two form elements. The first assesses the UNAMBIGUITY in the generated utterance, ensuring that it entails only the target slot value. The second assesses the WORLD-UNDERSTANDING criterion, leveraging 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.

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.[4] 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. To split INDIRECTREQUESTS into train, validation, and test sets, we divide the samples based on the 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.

5 Proxy Evaluation of Linguistic Criteria

We automate a proxy evaluation for IURs generations due to the impracticality of manual evaluation of numerous samples and models. This section defines the proxy evaluation task formulations and presents baseline results using zero-shot and few-shot prompting strategies. We define two proxy evaluation tasks, corresponding to UNAMBIGUITY and WORLD-UNDERSTANDING 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). Table 2 shows the performance of the proxy evaluators on the test split against the ground truth obtained through crowdsourcing.

[4] URL hidden for peer review.

Criterion	Model					
	Small LM (<1B)	GPT (3-shot) GPT-3.5	GPT-4	Llama 2 (3-shot) 7B	13B	70B
UNAMBIGUITY (Accuracy)	0.35* (nli-deberta)	0.73	0.84†	0.5	0.69‡	0.22
WORLD-UNDERSTANDING (Pearson correlation)	0.22* (ms-marco)	0.15	0.34†	0.16	0.19‡	0.18

Table 2: Evaluation results are computed from a single run with proxy evaluators against crowdworker annotations on the test split of INDIRECTREQUESTS, which contains 388 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, and ‡ signifies the top performer among the Llama 2 models (Touvron et al., 2023).

5.1.1 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[5] 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[6], fine-tuned on MS MARCO for passage ranking. We concatenate the IUR with the knowledge context, score the sequence using the model, and consider the IUR knowledgeable if the score exceeds 0.5.

5.1.2 Proprietary LLMs

For the proprietary LLMs from OpenAI, we use the models in a few-shot setup, providing a few examples of IURs labeled 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 knowledge context (if applicable) and take the model's output as the prediction.

5.1.3 Open-Source LLMs

For the open-source Llama 2 models, we use a similar few-shot setup as the proprietary LLMs. However, we also experiment with prompting the model with additional context, such as providing explicit instructions or examples tailored to the specific evaluation criterion. The results, shown in Table 2, highlight the trade-offs between model size, performance, and the ability to leverage additional context or prompting.

The prompts used for proprietary and open-source LLM based proxy evaluators is given 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 Table 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 develop a richer understanding of the specific "indirection strategies" that the LLMs employ to transform the slot schema into IURs. During analysis, one of the authors excluded those 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 3). **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

[5] nli-deberta-v3-small

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

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

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

In addition to carrying out automated, intrinsic evaluations, we also extrinsically evaluate the performance of a widely-used DST model over INDIRECTREQUESTS by measuring its drop in performance as compared to its performance on the SGD dataset. Since the DST model we use is trained on context window lengths of 3, the dialogue contexts in all samples are also set to 3. Table 4 shows a comparison between the model performance over the original samples and the samples using the generated IURs based on a total of 375 samples.

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)

8 Related Work

Brittleness of DST Models. The initiative to develop the IUR generation task springs from a need to reduce the brittleness of DST models. Cho et al. (2022) empirically demonstrate the brittleness of commonly used 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.

	SGD	INDIRECTREQUESTS
DST acc.	0.512	0.133

Table 4: Slot accuracies are computed for a T5-based state-of-the-art dialogue state tracking model on samples from both the original SGD dataset and the INDIRECTREQUESTS. The DST model performance on INDIRECTREQUESTS shows a significant degradation.

Relationship of IUR Generation to Other NLP Tasks. The process of generating IURs bears resemblance to paraphrase generation (Zhou and Bhat, 2021) in the aspect of semantically preserving text transformation. IUR generation also shares an inverse conceptual similarity with the NLI task of inferring entailment from a premise and a hypothesis. In contrast, IUR generation can be thought of as generating an NLI premise given a hypothesis and a positive entailment class. Although Shen et al. (2018) explore this very task formulation, their work differs significantly from ours as it is not situated in a dialogue context.

Text Generation using Small LLMs. Our research also investigates the impact of model size on the quality of the generated IURs. Eldan and

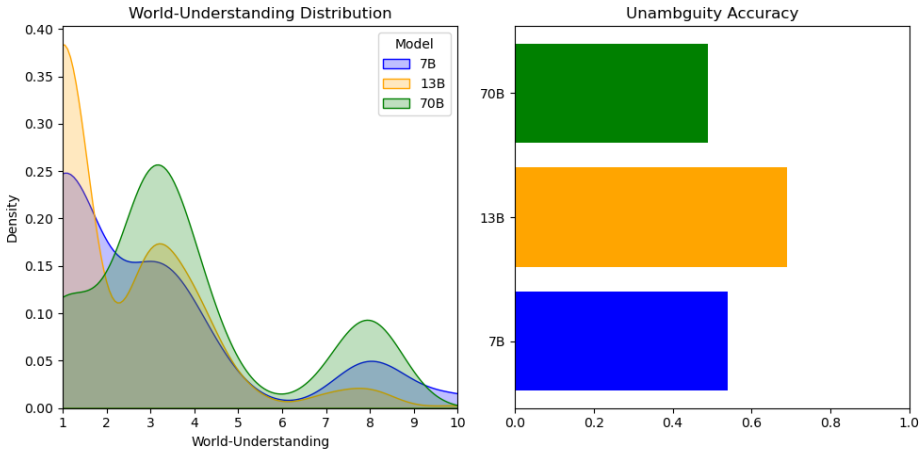

Figure 5: We report the qualities of the IURs generated using smaller, open-source Llama 2 models of three different sizes (7B, 13B, 70B). All the evaluation results are obtained using the best-performing GPT-4 proxy evaluation model (as described in Section 5).

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. By assembling 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 Discussion

The emergence of powerful LLMs in recent years (Brown et al., 2020) has led to near-perfect performances on several longstanding NLP benchmark datasets. As a result, the field of NLP has seen a shift from focusing solely on reporting quantitative performance metrics on benchmark datasets to conducting deeper qualitative analyses. Our work carries forward this trend by isolating the concept of indirectness in task-oriented dialogue utterances in the form of a dedicated benchmark dataset.

10 Limitations and Future Work

Our proposed notion of IUR applies only to categorical slots with a small, fixed number of possible values (< 5), but not to those slots that can take on a large number of values. Future work can investigate the IUR generation task for such challenging dialogue schemas. We have also limited ourselves to supervised fine-tuning of LLMs. However, there is a rich literature on the use of reinforcement learning to guide language models towards specific text styles and content types, especially for abstract concepts of the likes of *indirectness*, which can be explored as future work (Kaufmann et al., 2023). As Bowman and Dahl (2021) suggest, the ultimate evaluation measure for any NLP task should be grounded in in carefully annotated real user data. While modeling specific phenomena such as indirectness is helpful, the community needs to evolve novel evaluation paradigms in the long run. Until then, works such as ours will continue to inform gaps in existing models.

11 Conclusion

As the research and development of task-oriented dialogue systems advances, there is a pressing need to bridge the gap between benchmark corpora and utterances "in the wild." In our study, we concentrate on the phenomenon of "indirectness." This occurs when a user conveys their desired outcome in a manner that requires the listener to utilize their general knowledge to deduce the intended value. We develop a multi-stage LLM-based pipeline to generate INDIRECTREQUESTS, a dataset of IURs based on the schemas borrowed from the SGD dataset. INDIRECTREQUESTS supplements existing benchmarks to evaluate NLU and DST models on realistic, indirect user requests lacking explicit slot values. Experiments with a state-of-the-art DST model validate the challenging nature of INDIRECTREQUESTS. More broadly, our benchmark dataset can support future efforts for tasks such as API prediction, DST, NLU, which can lead to an overall improvement in the usability of virtual assistants for end users.

References

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*.

Vevake Balaraman, Seyedmostafa Sheikhalishahi, and Bernardo Magnini. 2021. Recent neural methods on dialogue state tracking for task-oriented dialogue systems: A survey. In *Proceedings of the 22nd annual meeting of the special interest group on discourse and dialogue*, pages 239–251.

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.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 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*.

Bill Byrne, Karthik Krishnamoorthi, Chinnadhurai Sankar, Arvind Neelakantan, Daniel Duckworth, Semih Yavuz, Ben Goodrich, Amit Dubey, Andy Cedilnik, and Kyu-Young Kim. 2019. Taskmaster-1: Toward a realistic and diverse dialog dataset. *arXiv preprint arXiv:1909.05358*.

Hyundong Cho, Chinnadhurai Sankar, Christopher Lin, Kaushik Ram Sadagopan, Shahin Shayandeh, Asli Celikyilmaz, Jonathan May, and Ahmad Beirami. 2021. CheckDST: Measuring real-world generalization of dialogue state tracking performance. *arXiv preprint arXiv:2112.08321*.

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].

Philip R Cohen. 2019. Foundations of collaborative task-oriented dialogue: what's in a slot? In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 198–209.

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

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

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*.

Shikib Mehri, Mihail Eric, and Dilek Hakkani-Tur. 2020. Dialoglue: A natural language understanding benchmark for task-oriented dialogue. *arXiv preprint arXiv:2009.13570*.

Johannes EM Mosig, Shikib Mehri, and Thomas Kober. 2020. Star: A schema-guided dialog dataset for transfer learning. *arXiv preprint arXiv:2010.11853*.

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.

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 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`