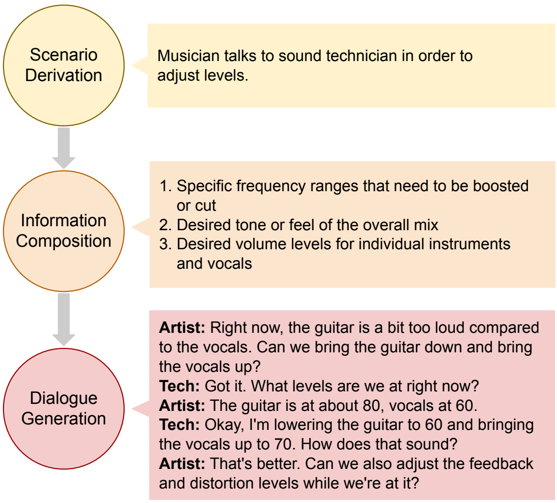
Dialogue State Generation: Transcending Slot Schemas for Domain-General State Inference

James D. Finch and **Boxin Zhao** and **Jinho D. Choi**
Department of Computer Science
Emory University
Atlanta, GA, USA
{jdfinch, zinc.zhao, jinho.choi}@emory.edu

1 Introduction

Task-oriented dialogue systems, which enable users to achieve specific goals via conversation, rely heavily on dialogue state tracking. Dialogue state tracking (DST) is the task of maintaining a structured representation of all goal-related information shared during the dialogue (Heck et al., 2023; Wang et al., 2023). Conventionally, a predefined slot schema specifies the types of information (slots) to be tracked. Although DST data provides a slot schema for each of several task domains, defining a good slot schema for a task-oriented dialogue system application is often nontrivial and time-consuming in practice (Rastogi et al., 2020; Budzianowski et al., 2018). Consequently, applications of DST models require substantial labor to create a slot schema, and will fail to cover important information if the schema is incomplete. Slot Discovery models have been proposed in the past to automatically induce slots from unlabeled dialogue data in a particular domain (Wu et al., 2022; Yu et al., 2022), but such data is often unavailable.

To address these limitations, we propose Dialogue State Generation (DSG), a task for inferring the state of a dialogue without access to domain-specific resources such as slot definitions or additional dialogue data. Formally, given a single dialogue history $D_{1..t}$ from turn 1 to t, DSG aims to produce a set of slot-value pairs $S_t = \{(s_1, v_1), (s_2, v_2), ..., (s_k, v_k)\}$, which represents all information in $D_{1..t}$ relevant to either speaker's goals. This task is challenging because the model must infer what information values are important, as well as slot names representing each type of information. Due to its flexibility to infer new slots on-the-fly, we believe DSG can assist or automate slot schema development, enhance traditional DST state predictions with additional slot-values, or directly infer dialogue states in a slot-schema-free dialogue system.



Musician talks to sound technician in order to adjust levels.

1. Specific frequency ranges that need to be boosted or cut
2. Desired tone or feel of the overall mix
3. Desired volume levels for individual instruments and vocals

Artist: Right now, the guitar is a bit too loud compared to the vocals. Can we bring the guitar down and bring the vocals up?
Tech: Got it. What levels are we at right now?
Artist: The guitar is at about 80, vocals at 60.
Tech: Okay, I'm lowering the guitar to 60 and bringing the vocals up to 70. How does that sound?
Artist: That's better. Can we also adjust the feedback and distortion levels while we're at it?

Figure 1: Examples of the three stages in the data generation pipeline used to create DSG5K.

2 DSG5K: New Diverse DSG Dataset

To train an end-to-end DSG model capable of generalizing to any domain to infer appropriate slots and values, a dialogue dataset with diverse slot-value labels is needed. However, existing data has limited domain and slot diversity; the most diverse dataset, Schema Guided Dialogues (SGD), covers 16 domains and 214 slot types (Rastogi et al., 2020). To overcome this challenge, we present the new DSG5K dataset, with 5,015 dialogues and 1,003 domains. We create DSG5K dialogues in three stages, shown in Figure 1. Initially, domains are derived through an iterative process of generating and de-duplicating dialogue scenario descriptions. Next, an unstructured list of information types associated with every scenario is generated. Finally, a dialogue is generated based on the scenario description and the unstructured information list. After dialogues are created, our LLM-based DSG pipeline GPTPipe (§3) annotates DSG5K with silver labels, producing a dataset for training DSG models with 173,572 unique slot names.

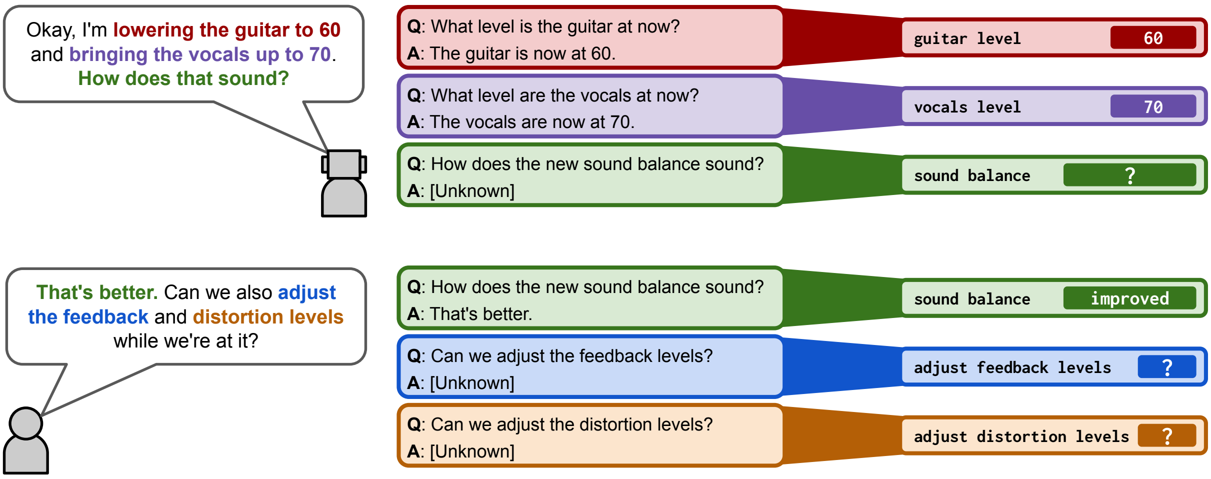

Okay, I'm lowering the guitar to 60 and bringing the vocals up to 70. How does that sound?
Q: What level is the guitar at now?
A: The guitar is now at 60.
guitar level
60
Q: What level are the vocals at now?
A: The vocals are now at 70.
vocals level
70
Q: How does the new sound balance sound?
A: [Unknown]
sound balance
?
That's better. Can we also adjust the feedback and distortion levels while we're at it?
Q: How does the new sound balance sound?
A: That's better.
sound balance
improved
Q: Can we adjust the feedback levels?
A: [Unknown]
adjust feedback levels
?
Q: Can we adjust the distortion levels?
A: [Unknown]
adjust distortion levels
?

Figure 2: Two example inferences of GPTPipe, a GPT-based DSG pipeline that infers dialogue state information by first generating question-answer pairs for important information shared in a given dialogue turn, then translates questions to slot names and answers to values to obtain the final dialogue state update.

3 GPTPipe: DSG with Zeroshot Pipeline

Given the exceptional zero-shot performance of instruction-tuned large language models (LLMs) on a wide variety of tasks (Brown et al., 2020; Kojima et al., 2022; Heck et al., 2023), we explore their ability in DSG. Specifically, we develop a pipeline that uses LLMs to automatically annotate every dialogue turn with a state update in two stages. First, *Question-Answer (QA) Pair Generation* deduces the key information in each turn by summarizing the turn's content as a list of QA pairs using GPT-4. Second, *Slot-Value Translation* translates those QA pairs into the corresponding slot names and values using GPT-3.5. Figure 2 illustrates the full GPTPipe pipeline.

4 Experiments

Data Our experiments are based on two datasets: Schema-Guided Dialogues (Rastogi et al., 2020), and DSG5K, our new dataset. For SGD, only the four held-out domains were used in the test set: *Alarm*, *Trains*, *Messaging*, and *Payment*. Furthermore, to improve slot diversity, we incorporated the Schema Guided Dialogues Extension (Lee et al., 2022). For DSG5K, we held out 100 domains to serve as evaluation data.

Models Three models were evaluated: SGD-DSG, E2E-DSG, and GPTPipe. SGD-DSG is a T5-3B model fine-tuned on the SGD dataset. E2E-DSG, is a T5-3B model fine-tuned on DSG5K with silver labels from GPTPipe. Lastly, we assessed the zero-shot performance of GPTPipe.

Evaluation Since DSG is slot-schema-free, we evaluate two aspects of DSG performance using binary human judgements. First, *Completeness* is the percentage of inferred state updates that cover all important information in the corresponding dialogue turn. *Correctness* represents the percentage of slot-value inferences that are accurate to the turn's content. Three human judges were recruited to annotate 60 turns from each evaluation dataset with these labels. Results are shown in Table 1.

	DSG5K			SGD		
Model	CP	CR	HM	CP	CR	HM
SGD-GSD	32.3	72.6	44.7	69.3	**90.8**	78.6
E2E-GSD	**95.7**	81.2	**87.9**	**94.7**	81.7	**87.7**
GPTPipe	93.3	**82.0**	87.3	90.0	84.7	87.3

Table 1: Human evaluation of SGD-DSG, E2E-DSG, and GPTPipe on DSG5K and SGD for *Completeness* (CP), *Correctness* (CR), and their harmonic mean (HM).

The evaluation found E2E-DSG to outperform the other models across both DSG5K and SGD datasets. Conversely, SGD-DSG struggled in *Completeness* due to poor diversity in its training data.

Conclusion This work introduced DSG, and demonstrates that it is possible to infer dialogue states in unseen task domains, even without a slot schema or other in-domain resources. The presented experiments also show that existing DST resources are currently insufficient for DSG, as SGD-DSG was unable to adapt to new domains. Our E2E-DSG model is the first to provide robust, slot-schema-free dialogue state inferences at low cost.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Pawe\l Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - A Large-Scale Multi-Domain Wizard-of-Oz Dataset for Task-Oriented Dialogue Modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Michael Heck, Nurul Lubis, Benjamin Ruppik, Renato Vukovic, Shutong Feng, Christian Geishauser, Hsien-Chin Lin, Carel van Niekerk, and Milica Gašić. 2023. ChatGPT for Zero-shot Dialogue State Tracking: A Solution or an Opportunity? ArXiv:2306.01386 [cs].

Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large Language Models are Zero-Shot Reasoners. *Advances in Neural Information Processing Systems*, 35:22199–22213.

Harrison Lee, Raghav Gupta, Abhinav Rastogi, Yuan Cao, Bin Zhang, and Yonghui Wu. 2022. SGD-X: A Benchmark for Robust Generalization in Schema-Guided Dialogue Systems. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(10):10938–10946. Number: 10.

Archiki Prasad, Peter Hase, Xiang Zhou, and Mohit Bansal. 2023. GrIPS: Gradient-free, Edit-based Instruction Search for Prompting Large Language Models. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3845–3864, Dubrovnik, Croatia. Association for Computational Linguistics.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards Scalable Multi-Domain Conversational Agents: The Schema-Guided Dialogue Dataset. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8689–8696. Number: 05.

Qingyue Wang, Liang Ding, Yanan Cao, Yibing Zhan, Zheng Lin, Shi Wang, Dacheng Tao, and Li Guo. 2023. Divide, Conquer, and Combine: Mixture of Semantic-Independent Experts for Zero-Shot Dialogue State Tracking. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2048–2061, Toronto, Canada. Association for Computational Linguistics.

Yuxia Wu, Lizi Liao, Xueming Qian, and Tat-Seng Chua. 2022. Semi-supervised New Slot Discovery with Incremental Clustering. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6207–6218, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Dian Yu, Mingqiu Wang, Yuan Cao, Izhak Shafran, Laurent Shafey, and Hagen Soltau. 2022. Unsupervised Slot Schema Induction for Task-oriented Dialog. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1174–1193, Seattle, United States. Association for Computational Linguistics.

A Error Analysis

An error analysis was conducted for each of our evaluated models by randomly sampling 100 turns per model from the human evaluation on DSG5K that were indicated to have at least one correctness error. One of the authors manually annotated each predicted slot-value pair in these sampled error cases with an error category, creating new categories as needed. Results and examples of this error analysis are shown in Table 2. The most significant error was hallucinating new information not present in the focal dialogue turn. We also observe that `E2E-DSG` errors are more likely to co-occur in the same turn compared to `GPTPipe`, resulting in more overall errors categorized for `E2E-DSG` in the error analysis despite achieving similar performance to `GPTPipe` in the evaluation.

B Prompts

Eliciting high-quality generations from an LLM on a particular task requires finding a suitable prompt. The prompt is the token sequence input to the LLM that includes both task-specific instructions and a formatted linearization of all inputs needed to complete one task sample. Searching for a prompt that maximizes task performance can be done manually or using automatic or semi-automatic search methods (Prasad et al., 2023). For complex tasks, multiple prompts can be used that decompose the task into more manageable subtasks. Due to the exploratory nature of our initial investigation into DSG, we develop prompts through a manual development process where generations are hand-checked for quality. This allows us to quickly try different strategies for writing prompt instructions and breaking DSG into subtasks.

DSG5K Generating information-rich dialogues is broken down into three stages: scenario derivation, information composition, and dialogue generation. The prompts for these stages are in Figures 3, 4, and 5, respectively.

GPTPipe For the development of GPTPipe, DSG is decomposed into two subtasks in a pipeline, *Question-Answer (QA) Pair Generation* to deduce the key information in each turn and *Slot-Value Translation* to transform those QA pairs into the corresponding slot names and values. It is noteworthy that we investigated several alternative strategies, which yielded poor performance. For example, directly generating state updates from a dialogue performed inadequately, even after numerous prompt revisions. The final list of prompts employed in GPTPipe are shown in Figures 6 - 9.

Error	Definition	Example Turn with Predicted State Update	SGD-DSG	GPT-Pipe	E2E-DSG

Hallucination	The slot-value represents information that is not mentioned or implied by the corresponding turn at all.	Okay, we'll definitely need to avoid peanuts and lower your sodium intake to manage your blood pressure. Are you currently physically active?
		`activity level: ?`
		`is alcohol served: ?`
			29.0	25.1	34.1

Leakage	The slot-value pair represents information that is only attributable to a previous turn in the dialogue, instead of being derivable from the focal turn.	Alright, Sarah. Can you tell me which payroll period the issues occurred in?
		`payroll period: ?`
		`name: Sarah Johnson`
		`employee ID: 123456`
			2.8	16.0	26.5

Inaccurate	The slot-value pair poorly represents the intended information in the focal turn.	Sure, let's have a sectional rehearsal for that. Bass and cello sections, please come forward. I noticed that in measure 100, some of you were playing a bit too loudly. Please remember to keep the dynamics level until measure 112.
		`beginning note of song: 100`
		`end of song: 112`
			30.8	6.3	12.1

Partial	The value for the particular slot only partially captures the full meaning presented in the focal turn and is thus incomplete.	I believe the target audience for the yoga article would be pregnant women and new mothers, while the sugar article would appeal to a broader group interested in health and wellness. **`audience type: Health & Wellness`**	12.1	3.3	2.8

Redundant	The slot-value pair represents the same information as another slot-value pair for the focal turn.	My name is John Smith and you can reach me at 555-1234.
		`name: John Smith`
		`contact info: 555-1234`
		`first name: John`
		`last name: Smith`
		`phone number: 555-1234`
			0.9	19.1	12.4

Imprecise	The slot-value representation is unclear and ambiguous in meaning when taken on its own rather than clearly and concisely summarizing some presented information from the focal turn.	When I was growing up, my parents always taught me to appreciate nature and not take it for granted. We used to go on camping trips and nature walks, and those memories stay with me even today.
		`parents teachings: Appreciate nature`
		`camping trips: Yes`
		`nature walks: Yes`
		`camping nature memories: Yes`
			0.9	9.8	15.5

Missing	Some information that is presented in the focal turn and is necessary for capturing its full meaning is not represented by any slot-value pair for the turn.	Well, my current schedule is Monday through Friday, 8:00 am to 5:00 pm. I was thinking about changing it up a bit.
		`day of appointment: Monday`
		`end of the available event: 5:00 pm`
			27.1	0.0	0.0

Good	No errors		30.8	34.7	26.5
Total slot-value pairs predicted			107	398	355

Table 2: Error analysis of 100 predicted state updates per model, where each state update contains at least one correctness error identified by human judges during evaluation. Analysis results show the percentage of slot-value pairs belonging to various error categories, but categories are not mutually exclusive. Example turns are abbreviated from real outputs, and the slot-value pairs that express the error type are bolded.

```
List 100 diverse examples of everyday tasks that require talking to another person.
Format each list item like:

N. <Role of person 1> talks to <role of person 2> in order to <task goal>
```

Figure 3: GPT-3.5 prompt for generating dialogue scenarios/domains.

```
List examples of as many different types of information as you can that would be
shared during the dialogue scenario: {domain}
```

Figure 4: GPT-3.5 prompt for generating a list of information types for each dialogue domain.

```
Dialogue Scenario:
{domain}

Information Types:
{info types}

Write a dialogue for the above Dialogue Scenario. Include specific examples of the
Information Types above being shared and implied throughout the conversation.
Make up actual names/values when specific information examples are shared.
```

Figure 5: GPT-3.5 prompt for generating a dialogue for a given task domain.

```
Two people, {speaker} and {listener}, are having a dialogue in which the
following was just said:

{dialogue context}
{speaker}: {last turn}

Please break down and summarize all the information in what {speaker} just
said into as many question-answer pairs as you can. Each question-answer pair
should be short, specific, and focus on only one piece of information or value.

For information {speaker} shared, use the question-answer pair format:

{listener}: <question>
{speaker}: <answer>

For information {speaker} requested or indicated not knowing,
use the answer "Unknown." in a question-answer pair format like:

{speaker}: <question>
{listener}: Unknown.


{answered qa pairs}
```

Figure 6: GPT-4 prompt for generating question-answer pairs for a dialogue context.

```
Two people, {speaker} and {listener}, are having a dialogue in which the
following was just said:

{dialogue context}
{speaker}: {last turn}

Please identify the information or values {speaker} gave as short answers to the
following questions (use the answer "Unknown." if the question is not answered by
{speaker} in the dialogue):

{unanswered qa questions}
```

Figure 7: GPT-4 prompt for answering questions from the previous turn that were not previously answered.

```
{qa pairs}

Translate each question above into variable names.
Each label should be very short, usually one or two words,
but specific to the details of the question. Write each question before
translating it into a variable name, in the format:

<question> -> <variable name>
```

Figure 8: GPT-3.5 prompt for translating questions into slot names.

```
{qav tuples}

Translate each answer to the above questions into a value for the
corresponding variable. Values should be short, usually one word,
very short phrase, number, span, category, score, boolean, list,
or other value. Copy each answer before translating it into a value,
in the format:

Question: <question>
Variable: <variable>
Answer: <answer>
Value: <value>
```

Figure 9: GPT-3.5 prompt for translating answers into slot values.