

Generative Dialogue State Tracking with GPT-2 for Task-Oriented Service Conversations

Stanford CS224N Custom Project

Venu Samprathi
Department of Computer Science
Stanford University
samprath@stanford.edu

March 12, 2026

Abstract

Dialogue state tracking (DST) is a critical component of task-oriented dialogue systems, requiring models to maintain a structured representation of user goals across multi-turn conversations. SimpleTOD demonstrated that a single autoregressive language model (GPT-2) can perform DST by casting belief state prediction as sequence generation. In this work, I replicate SimpleTOD on MultiWOZ 2.4 and investigate whether domain-adaptive pretraining (DAPT), continuing GPT-2’s pretraining on the Schema-Guided Dialogue (SGD) corpus before fine-tuning, improves downstream DST accuracy. With GPT-2 small (124M), my baseline achieves 59.00% joint goal accuracy (JGA), exceeding the published SimpleTOD benchmark of 57.18%, while the DAPT model achieves only 57.87% (−1.13). Scaling to GPT-2 medium (355M) improves both models (baseline 60.48%, DAPT 60.44%), narrowing the gap to −0.04 points. Quantitative Jaccard analysis reveals near-zero slot name overlap between SGD and MultiWOZ ($J=0.0$ for 4 of 5 domains), explaining why DAPT fails to transfer useful slot-level representations. These results suggest that when sufficient in-domain fine-tuning data is available, continued pretraining on a schema-mismatched corpus does not help, and that increased model capacity is a more reliable path to improved DST accuracy.

1 Introduction

Task-oriented dialogue systems power virtual assistants that help users accomplish goals such as booking restaurants, finding trains, or reserving hotels through natural conversation. At the core of these systems is dialogue state tracking (DST): at each turn, the system must maintain a structured belief state, a set of (domain, slot, value) triplets, representing what the user wants. Accurate DST is essential because all downstream actions (database queries, API calls, response generation) depend on correctly identifying user constraints.

SimpleTOD [1] showed that GPT-2 can perform DST by treating it as autoregressive sequence prediction, using special delimiter tokens to encode belief states as flat text. A natural question arises: can we improve this by giving the model more exposure to task-oriented dialogue before fine-tuning? Gururangan et al. [2] demonstrated that domain-adaptive pretraining (DAPT) improves performance on downstream classification tasks. If the same principle applies to generative DST, a GPT-2 model further pretrained on dialogue text should achieve higher accuracy when fine-tuned for belief state prediction.

I test this hypothesis by continuing GPT-2’s pretraining on the Schema-Guided Dialogue (SGD) dataset [3] (over 20,000 dialogues across 20 domains), then fine-tuning on MultiWOZ 2.4 [4] for DST. The results are a negative finding: DAPT does not help. With GPT-2 small (124M), the baseline achieves 59.00% JGA while the DAPT model achieves only 57.87% (−1.13). Scaling to GPT-2 medium (355M), the baseline improves to 60.48% and DAPT to 60.44% (−0.04). Per-domain analysis shows that DAPT degrades performance most where slot schemas diverge, and Jaccard analysis reveals near-zero slot name overlap between SGD and MultiWOZ ($J=0.0$ for 4 of 5 domains). I analyze several explanations including schema-level mismatch, special token embedding disruption, and the interaction between model capacity and domain adaptation.

2 Related Work

Dialogue state tracking. DST has evolved from rule-based systems to neural approaches. TRADE [5] introduced a copy mechanism to generate slot values from dialogue context. TripPy [6] proposed a triple copy strategy using span prediction and cross-slot coreference. These modular approaches treat DST as a standalone component, separate from response generation.

Language models for dialogue. SimpleTOD [1] unified DST, action prediction, and response generation into a single autoregressive sequence using GPT-2, achieving 57.18% JGA on MultiWOZ 2.4 [4], outperforming TRADE’s 55.05%.

Domain-adaptive pretraining. Gururangan et al. [2] showed that continuing pretraining on domain-relevant text improves performance across eight classification tasks, with the largest gains when the pre-training corpus matched the target domain’s distribution.

Schema-Guided Dialogue. The SGD dataset [3] provides over 20,000 dialogues across 20 service domains. While SGD and MultiWOZ share some domain overlap (restaurants, hotels, transit), they differ in annotation schemes, slot naming conventions, and dialogue collection methodology.

3 Approach

3.1 Task Formulation

Following SimpleTOD, I formulate DST as autoregressive sequence prediction. At each dialogue turn t , the model receives the conversation history and generates the cumulative belief state. The input-output format uses special delimiter tokens:

```
<|context|> <|system|> ... <|user|> ... <|endofcontext|> <|belief|> domain slot value , ...  
                                <|endofbelief|>
```

The model is trained with cross-entropy loss over the entire sequence. At inference, it receives everything up to <|belief|> and generates autoregressively until producing <|endofbelief|>.

3.2 Data

MultiWOZ 2.4 [4] contains approximately 10,000 dialogues across 7 domains with corrected annotations. DST evaluation covers 5 domains: attraction, hotel, restaurant, taxi, and train. I use the standard splits: 8,438 training dialogues (56,668 turn-level samples), 1,000 validation (7,374 samples), and 1,000 test (7,368 samples).

Schema-Guided Dialogue (SGD) [3] contains over 20,000 dialogues across 20 service domains. For DAPT, I use all splits as raw text, stripping annotations and retaining only speaker labels and utterances (~22,000 dialogues).

Preprocessing. For MultiWOZ, each turn is linearized into a tokenized sequence following the SimpleTOD format, with a sliding context window within GPT-2’s 1024-token limit. For SGD, dialogues are concatenated with speaker labels, chunked at 1024 tokens, and tokenized using GPT-2’s BPE tokenizer without special tokens.

3.3 Models

I experiment with GPT-2 small (124M) and GPT-2 medium (355M) [7]. For both, six special tokens are added (<|context|>, <|endofcontext|>, <|user|>, <|system|>, <|belief|>, <|endofbelief|>), with new embeddings initialized from a multivariate normal distribution matching the existing embeddings’ mean and covariance.

Baseline. GPT-2 is fine-tuned on MultiWOZ 2.4 for 10 epochs with AdamW ($\text{lr} = 5 \times 10^{-5}$, effective batch size 32). GPT-2 medium uses gradient accumulation (micro-batch 16×2 steps).

DAPT model. Stage 1: GPT-2 is pretrained on SGD for 5 epochs without special tokens. Stage 2: the DAPT checkpoint is fine-tuned on MultiWOZ 2.4 with special tokens added, using identical hyperparameters to the baseline.

Medium model improvements. GPT-2 medium experiments add: (1) cosine LR schedule with 10% warmup; (2) gradient norm clipping at 1.0; and (3) per-step logging of loss, gradient norms, and learning rate. These were applied uniformly to all medium experiments.

3.4 Evaluation Metrics

Joint Goal Accuracy (JGA): percentage of turns where the predicted belief state exactly matches ground truth across all active slots. **Per-domain JGA**: computed independently for each domain’s slots. **Slot-level F1**: treats each (domain, slot, value) triplet as a binary prediction, giving partial credit for partially correct turns.

4 Experiments

4.1 Training Details

All experiments ran on a single NVIDIA RTX Pro 6000 GPU (96 GB VRAM). Table 1 summarizes the training configurations.

Model	Stage	Epochs	Train Loss (final)	Eval Loss (best)	Schedule
GPT-2 small	DAPT (SGD)	5	0.5899	—	constant
	Baseline (MultiWOZ)	10	0.2539	—	constant
	DAPT FT (MultiWOZ)	10	0.2498	—	constant
GPT-2 medium	DAPT (SGD)	5	0.5547	0.5494	cosine
	Baseline (MultiWOZ)	10	0.1746	0.5414	cosine
	DAPT FT (MultiWOZ)	10	0.1815	0.5378	cosine

Table 1: Training loss progression. All use AdamW with $\text{lr} = 5 \times 10^{-5}$ and effective batch size 32. Medium adds cosine schedule with 10% warmup and gradient clipping at 1.0.

Both medium models show best eval loss at epoch 2 (0.5414 baseline, 0.5378 DAPT), with eval loss rising thereafter, indicating overfitting. However, downstream JGA continues improving past the eval loss minimum, suggesting that language modeling loss on padded sequences is an imperfect proxy for generation-based DST accuracy.

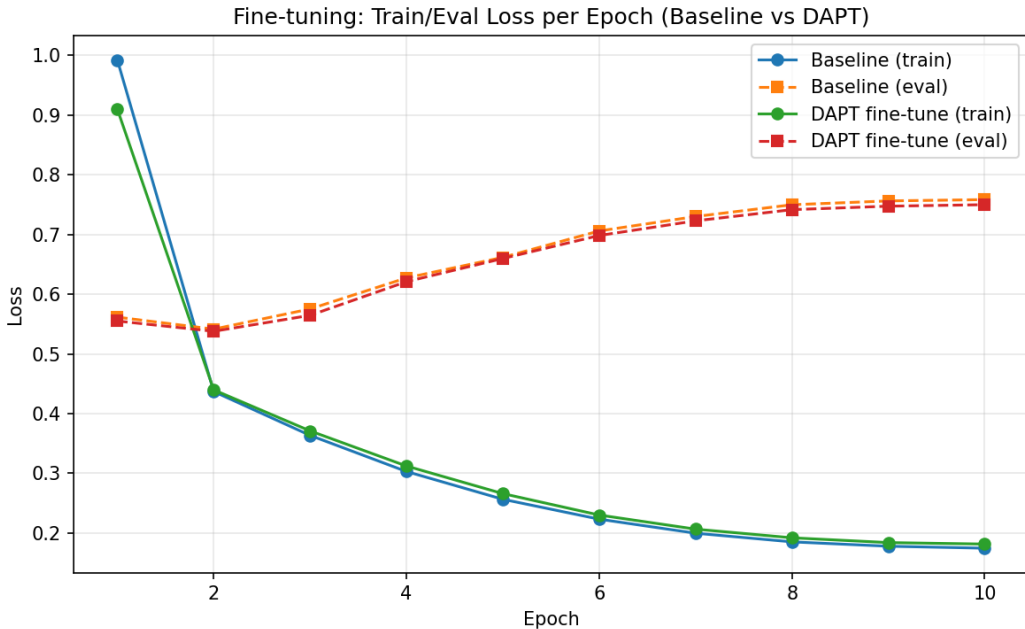


Figure 1: Train and eval loss per epoch for GPT-2 medium. Both models show nearly identical convergence: training loss decreases monotonically while eval loss bottoms at epoch 2. The DAPT model starts with slightly lower training loss but offers no eval loss advantage.

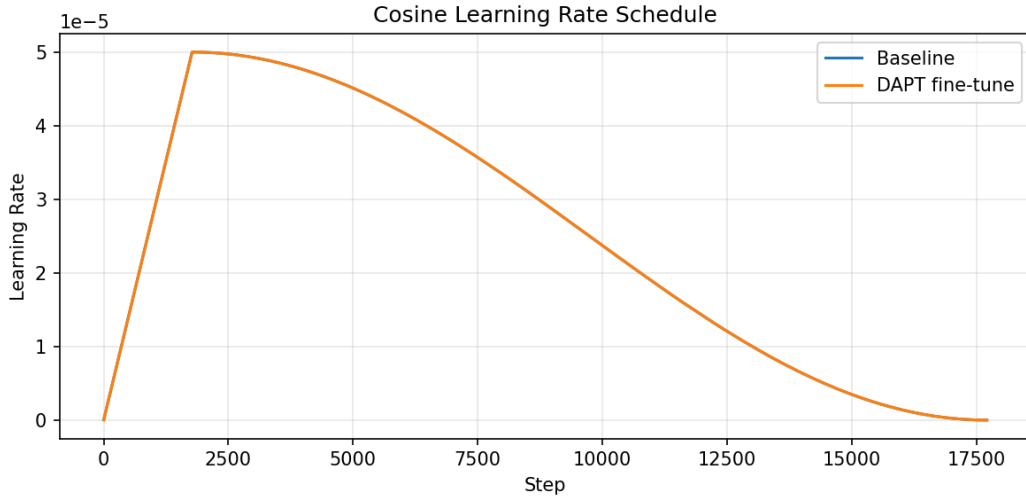


Figure 2: Cosine learning rate schedule with 10% warmup used for all GPT-2 medium experiments. Both baseline and DAPT fine-tuning use identical schedules (17,710 total optimizer steps), ensuring a fair comparison.

4.2 Main Results

Table 2 presents the primary results on the MultiWOZ 2.4 test set.

Model	Size	JGA (%)	Prec. (%)	Rec. (%)	F1 (%)
TRADE (published)	—	55.05	—	—	—
SimpleTOD (published)	—	57.18	—	—	—
Baseline	124M	59.00	93.25	93.09	93.17
DAPT (best, epoch 5)	124M	57.87	92.64	93.26	92.95
Baseline	355M	60.48	94.59	93.12	93.84
DAPT (epoch 10)	355M	60.44	94.31	93.22	93.76

Table 2: Test set results on MultiWOZ 2.4. Published scores from [4]. The DAPT gap narrows from -1.13 to -0.04 when scaling from 124M to 355M parameters.

My GPT-2 small baseline achieves 59.00% JGA, exceeding published SimpleTOD by 1.82 points, likely from training on corrected MultiWOZ 2.4 annotations. Scaling to medium yields: (1) both models improve (baseline $+1.48$, DAPT $+2.57$); and (2) the DAPT penalty nearly vanishes (-0.04 , within noise), suggesting increased capacity absorbs the distributional shift from DAPT (Section 5.2). The slot-level F1 difference between baseline and DAPT is small (-0.22 for small, -0.08 for medium), confirming that JGA failures arise from small errors compounding across multi-slot turns rather than catastrophic failures.

4.3 Per-Domain Analysis

Table 3 breaks down JGA by domain for both model sizes.

With GPT-2 small, DAPT hurts uniformly: taxi takes the largest hit (-3.10). With medium, the pattern shifts: DAPT is neutral or positive for 4 of 5 domains, but hotel remains an outlier at -1.78 . Hotel has the most complex slot schema in MultiWOZ (10 slots including booking fields), making it most sensitive to distributional shift. The taxi domain shows the most dramatic capacity-driven improvement (59.82% to 70.21%, $+10.39$), suggesting that low-resource domains benefit most from increased model size.

Quantitative slot overlap. Table 4 presents Jaccard similarity between SGD and MultiWOZ slot names per domain.

Even “strongly overlapping” domains like hotel and restaurant share *zero* slot names. SGD uses descriptive names (`check_in_date`, `price_per_night`) while MultiWOZ uses abbreviated conventions (`book`

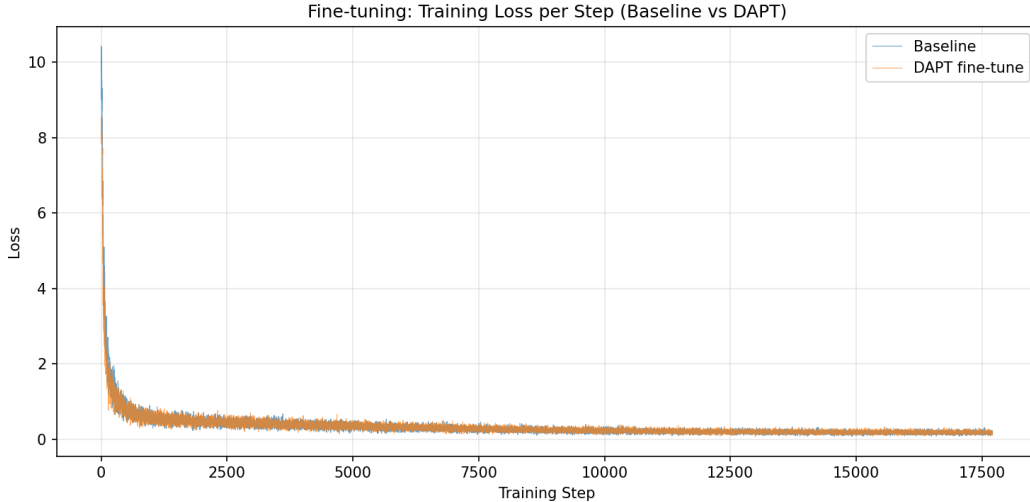


Figure 3: Per-step training loss during fine-tuning (GPT-2 medium). The DAPT model starts at a lower loss (benefiting from SGD pretraining) but both models converge to nearly identical trajectories by step $\sim 2,000$, showing that any initial DAPT advantage dissipates quickly.

Domain	GPT-2 small (124M)			GPT-2 medium (355M)		
	Base	DAPT	Δ	Base	DAPT	Δ
Attraction	71.08	69.73	-1.35	70.06	70.25	+0.19
Hotel	55.88	55.51	-0.37	56.55	54.77	-1.78
Restaurant	69.87	69.95	+0.08	73.25	73.42	+0.17
Taxi	59.82	56.72	-3.10	70.21	71.45	+1.24
Train	79.80	78.57	-1.23	80.73	81.13	+0.40
Overall	59.00	57.87	-1.13	60.48	60.44	-0.04

Table 3: Per-domain JGA on MultiWOZ 2.4 test set. With small, DAPT hurts nearly all domains. With medium, only hotel shows a notable DAPT penalty (-1.78).

day, pricerange). Since DAPT operates on raw text without annotations, it cannot learn the structural slot-value mappings that DST requires, explaining why DAPT fails to transfer useful representations.

5 Analysis

5.1 Why DAPT Did Not Help

The negative result contradicts the hypothesis that more dialogue exposure would improve representations. Three factors contribute:

Schema-level distributional mismatch. The Jaccard analysis (Table 4) confirms near-zero slot name overlap despite domain similarity. Furthermore, SGD spans 20 domains, 15 of which (Banks, Calendar, Media, Weather, etc.) have no MultiWOZ counterpart, diluting the pretraining signal.

Special token embedding disruption. DAPT uses raw text without special tokens. When $\langle |\text{belief}| \rangle$ and $\langle |\text{endofbelief}| \rangle$ are added for fine-tuning, their embeddings are initialized on weight matrices already shifted from original GPT-2 values. The baseline initializes on the original weights, potentially making it easier to learn delimiter token roles.

Sufficient fine-tuning data. MultiWOZ provides 56,668 turn-level samples, sufficient for fine-tuning even a 355M model. DAPT is most beneficial when downstream data is limited; with ample data, the model learns domain patterns directly, making intermediate pretraining redundant or harmful.

MultiWOZ	SGD Service(s)	Jaccard	Shared Slots
Attraction	Travel_1	0.000	$\emptyset$
Hotel	Hotels_1-4 (combined)	0.000	$\emptyset$
Restaurant	Restaurants_1-2 (combined)	0.000	$\emptyset$
Taxi	RideSharing_1-2 (combined)	0.125	{destination}
Train	Trains_1	0.000	$\emptyset$

Table 4: Jaccard similarity of slot names between SGD and MultiWOZ. Despite covering similar services, the datasets use almost entirely different slot naming conventions. Only taxi/RideSharing shares one slot name.

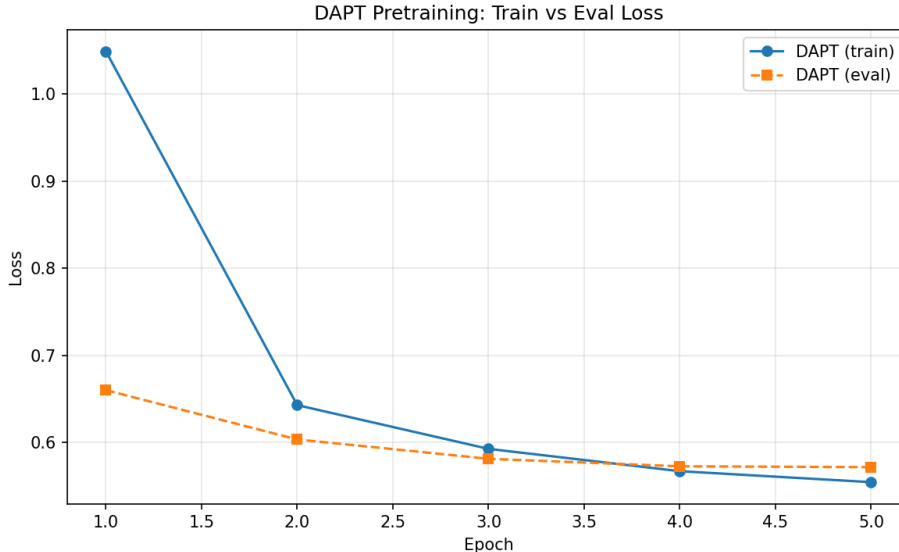


Figure 4: DAPT pretraining loss on SGD (GPT-2 medium). Train and eval losses converge by epoch 4–5, confirming the model learns the SGD distribution. However, this distribution has near-zero schema overlap with MultiWOZ (Table 4).

5.2 Effect of Model Capacity

Capacity improves absolute performance. Medium baseline achieves 60.48% vs. 59.00% for small (+1.48). The improvement is largest for taxi (+10.39), a low-resource domain that benefits most from the larger model’s generalization ability.

Capacity absorbs distributional shift. The DAPT penalty shrinks from -1.13 (small) to -0.04 (medium). With more parameters, the model can “overwrite” SGD-specific patterns during fine-tuning. This can be understood through a bias-variance lens: DST is a high-bias task (exact slot conventions required) with low variance (fixed schemas). DAPT introduces variance from a mismatched distribution without reducing bias. Larger models have lower bias to begin with, making them more robust to the added variance.

5.3 Comparison with Gururangan et al.

The original DAPT paper [2] showed consistent gains with RoBERTa on classification tasks. Key differences explain the divergence: (1) **Task type**: classification is more forgiving of distributional shift than exact structured generation; (2) **Domain alignment**: their corpora were carefully matched (CS papers for ACL-ARC), while SGD and MultiWOZ have near-zero schema alignment; (3) **Architecture**: RoBERTa (masked LM) vs. GPT-2 (autoregressive), which may interact differently with continued pretraining. This also explains the bias-variance connection: classification has lower bias requirements (coarser output space), making it more tolerant of distributional mismatch.

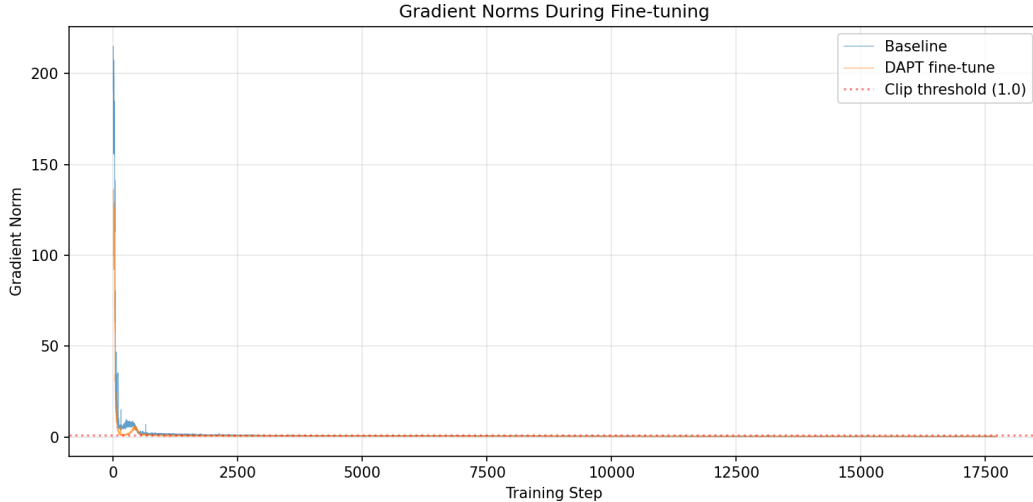


Figure 5: Gradient norms during fine-tuning (GPT-2 medium). Both models show large initial spikes (> 100) as the model adapts to new special token embeddings, then stabilize below the clipping threshold (1.0). The baseline spike is more pronounced since it learns special tokens from scratch.

5.4 Baseline Replication Quality

My GPT-2 small baseline (59.00% JGA) exceeds published SimpleTOD (57.18%) by 1.82 points, consistent with training on corrected MultiWOZ 2.4 annotations. The medium baseline further improves to 60.48%. Both validate the implementation and establish reliable baselines.

6 Conclusion

I replicated SimpleTOD’s generative DST approach on MultiWOZ 2.4, achieving 59.00% JGA with GPT-2 small and 60.48% with GPT-2 medium, both exceeding the published benchmark. DAPT on SGD degrades JGA by 1.13 points with the small model; the penalty nearly vanishes (-0.04) with medium. Jaccard analysis reveals near-zero slot name overlap, explaining the transfer failure.

Three conclusions emerge: (1) DAPT is not universally beneficial; when the pretraining corpus has schema-level mismatch, it can hurt; (2) increased model capacity (+1.48 JGA) is more reliable than domain pretraining; and (3) smaller models are more vulnerable to distributional shift from continued pretraining.

Ethical considerations. DST accuracy varies substantially across domains (train 80.73% vs. hotel 56.55%), meaning users with hotel queries receive less reliable service. Generative models may memorize training data containing user preferences, and MultiWOZ’s English crowdworker data does not represent diverse user populations.

Future work. Promising directions include: (1) low-resource DST where DAPT may help more; (2) schema-aligned pretraining using SGD data converted to SimpleTOD format; (3) mixed-data fine-tuning on both SGD and MultiWOZ; and (4) parameter-efficient methods like LoRA [8] that preserve more pretrained weights.

Acknowledgments

I thank my TA mentor Ahmed Ahmed for guidance and feedback throughout this project.

References

- [1] Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. Simpletod: A simple language model for task-oriented dialogue. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.
- [2] Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. Don’t stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2020.

- [3] Abhinav Rastogi, Xiaoxue Zang, Srinivas Sber, Raghav Gupta, and Pranav Shyam. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.
- [4] Fanghua Ye, Jarana Manotumruksa, Qiang Zhang, Shenghui Li, and Emine Yilmaz Lizio. Multiwoz 2.4: A multi-domain task-oriented dialogue dataset with essential annotation corrections. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, 2022.
- [5] Chien-Sheng Wu, Andrea Madotto, Ehsan Hosseini-Asl, Caiming Xiong, Richard Socher, and Pascale Fung. Transferable multi-domain state generator for task-oriented dialogue systems. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2019.
- [6] Michael Heck, Carel van Niekerk, Nurul Lubis, Christian Geishauser, Hsien-Chin Lin, Marco Moresi, and Milica Gašić. Trippy: A triple copy strategy for value independent neural dialog state tracking. In *Proceedings of the 21st Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, 2020.
- [7] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 2019.
- [8] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022.