GEM: Graph-Enhanced Mixture-of-Experts with ReAct Agents for Dialogue State Tracking

Ziqi Zhu*, **Adithya Suresh***, **Tomal Deb**†, **Iman Abbasnejad**†

Generative AI Innovation Center, Amazon Web Services
{ziqizhu, adxthya, tomalde, imanaba}@amazon.com

Abstract

Dialogue State Tracking (DST) requires precise extraction of structured information from multi-domain conversations, a task where Large Language Models (LLMs) struggle despite their impressive general capabilities. We present **GEM** (Graph-Enhanced Mixture-of-Experts), a novel framework that combines language models and graph-structured dialogue understanding with ReAct agent-based reasoning for superior DST performance. Our approach dynamically routes between specialized experts: a Graph Neural Network that captures dialogue structure and turn-level dependencies, and a fine-tuned T5-Small encoder-decoder for sequence modeling, coordinated by an intelligent router. For complex value generation tasks, we integrate ReAct agents that perform structured reasoning over dialogue context. On MultiWOZ 2.2, GEM achieves **65.19% Joint Goal Accuracy**, substantially outperforming end-to-end LLM approaches (best: 38.43%) and surpassing state-of-the-art (SOTA) methods including TOATOD (63.79%), D3ST (58.70%), and Diable (56.48%). Our graph-enhanced mixture-of-experts architecture with ReAct integration demonstrates that combining structured dialogue representation with dynamic expert routing and agent-based reasoning provides a powerful paradigm for dialogue state tracking, achieving superior accuracy while maintaining computational efficiency through selective expert activation.

Introduction

Dialogue State Tracking (DST) constitutes a fundamental component of task-oriented dialogue systems, responsible for accurately monitoring user intentions and extracting structured information throughout conversational interactions. As conversations progress across multiple turns and domains, DST systems must maintain precise representations of user goals, preferences, and constraints to enable appropriate system responses. Despite significant advances in neural architectures (Wu et al. 2019; Bebensee and Lee 2023; Lesci et al. 2023), achieving robust performance in complex multi-domain scenarios remains challenging due to the inherent difficulties in tracking evolving contextual information and resolving ambiguities across conversation turns.

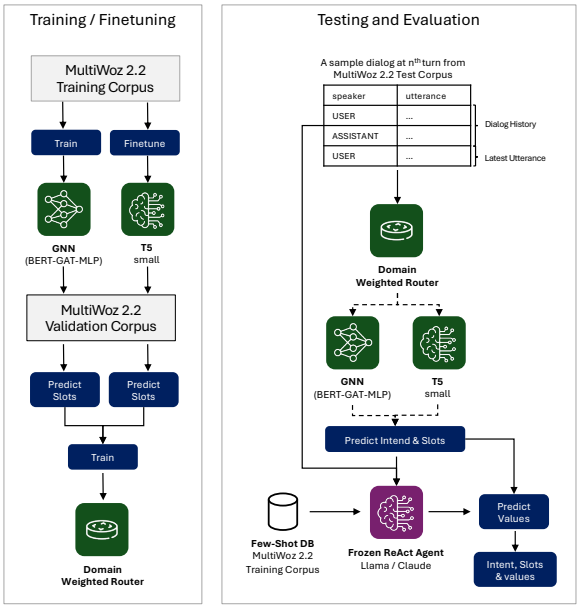

Figure 1: Our proposed architecture for dialogue state tracking. The MoE dynamically routes inputs between T5 encoder-decoder (Raffel et al. 2020) and GNN model via a learned gating network. This hybrid approach effectively combines the strengths of transformer-based language understanding with graph-based relational reasoning for improved intent detection, slot prediction, and value extraction in multi-turn conversations.

The development of large language models (LLMs) has significantly transformed natural language processing capabilities, demonstrating unprecedented performance across diverse tasks (Mann et al. 2020; Bang et al. 2023). Particularly noteworthy is their emergent reasoning ability, which enables complex problem-solving through stepwise decomposition of tasks (Wei et al. 2022; Huang and Chang 2022). The plan-and-solve paradigm (Chen et al. 2025) has further enhanced this capacity by prompting LLMs to generate explicit reasoning steps, allowing them to address intricate challenges through systematic deconstruction into manageable sub-tasks (Khot et al. 2022). However, as these models grow increasingly large and complex, they present substantial computational

*These authors contributed equally.
†These authors contributed equally.

challenges while still exhibiting fundamental limitations in knowledge representation and reasoning reliability. Despite their significant advancements, LLMs remain constrained by knowledge limitations and susceptibility to hallucinations during reasoning processes, leading to consequential errors in critical applications (Hong et al. 2023; Wang et al. 2023).

Knowledge graphs (KGs) offer a promising direction for addressing these limitations by providing structured factual representations that can enhance dialogue understanding (Pan et al. 2024). Existing integration approaches are typically divided between semantic parsing methods that convert questions into logical queries (Lan and Jiang 2020; Ye et al. 2021) and retrieval-augmented methods that enhance context with relevant information (Jiang et al. 2022). More sophisticated knowledge fusion techniques like retrieval-augmented generation (RAG)(Lewis et al. 2020) and knowledge graph prompting(Zhang et al. 2024) have emerged to address these limitations. However, these approaches face persistent challenges: semantic parsing often produces non-executable queries due to syntactic constraints, while retrieval-based methods typically fail to fully utilize the rich structural information inherent in knowledge graphs (Mavromatis and Karypis 2024). Furthermore, agent-based methods necessitate multiple interaction rounds, increasing computational overhead (Dehghan et al. 2024).

Computational efficiency presents another critical consideration for advanced dialogue systems. The Mixture-of-Experts (MoE) approach (Xue et al. 2024) offers a promising solution by activating only a subset of parameters or models at any given time, allowing the total parameter count to grow without proportional increases in computational requirements. LLMs scaled using this approach have demonstrated impressive performance across downstream tasks (Jiang et al. 2024), yet their application to structured dialogue state tracking remains underexplored.

This paper introduces a novel framework for DST that addresses these challenges through a Graph Enhanced Mixture-of-Experts (GEM) architecture. Our approach dynamically constructs and updates knowledge graphs through graph neural networks (GNNs), harnessing their representation learning capabilities to process complex structural information while overcoming their inherent limitations in natural language understanding. Different dialogue domains naturally exhibit distinct processing requirements — some benefit from structural reasoning while others require sequential linguistic understanding — motivating our dynamic expert routing approach. The framework employs a routing mechanism that dynamically allocates computational resources between a finetuned T5 encoder-decoder model (Raffel et al. 2020) and our proposed GNN-based method, activating only the most appropriate expert for each specific dialogue segment. Unlike conventional methods, our graph-based representation system captures the intricate, evolving relationships between dialogue states, user intents, and system responses throughout conversation turns. We further enhance the architecture through an agentic-based approach for value generation that leverages structured reasoning via the ReAct framework, significantly improving both the accuracy and interpretability of slot-value extraction in challenging multi-domain conversational scenarios. Figure 1 illustrates our proposed method.

In this paper we make the following contributions:

1. A fine-tuned graph attention network architecture for accurate intent detection and slot prediction that effectively models complex interactional dynamics in multi-turn conversations,
2. A routing mechanism that dynamically selects between finetuned T5 encoder-decoder and GNN expert models based on query characteristics, optimizing the balance between computational efficiency and prediction accuracy,
3. Extending the architecture by integration of a pre-trained decoder-LLM for robust, slot-conditioned value extraction,
4. Extending the solution through an agentic-based approach that leverages conversation history, content context, and identified slots to perform reasoning-based value generation,
5. Obtaining the SOTA results on the challenging MultiWOZ dataset, demonstrating the framework's effectiveness in complex dialogue systems.

Related work

DST is a critical component of task-oriented dialogue systems, requiring precise tracking of user intents and slot values throughout conversations. Traditional approaches have evolved from rule-based systems to neural network architectures, with sequence-based models like TRADE (Wu et al. 2019) and TripPy (Heck et al. 2020) establishing strong baselines. More recent innovations include DS-DST (Zhang et al. 2019), Seq2Seq-DU (Feng, Wang, and Li 2021), LUNA (Wang et al. 2022), SPACE-3 (He et al. 2022), SPLAT (Bebensee and Lee 2023), Diable (Lesci et al. 2023), D3ST (Zhao et al. 2022) and TOATOD (Bang, Lee, and Koo 2023). While these models have advanced performance boundaries, they continue to face limitations in modeling complex interactional dynamics across multi-turn dialogues, particularly in multi-domain scenarios where context flows between distinct conversational topics.

LLMs have transformed the DST landscape, demonstrating impressive general language understanding capabilities (Mann et al. 2020; Bang et al. 2023). However, they faces challenges in extracting structured information and maintaining consistency, leading to issues like hallucination (Hudeček and Dusek 2023; Heck et al. 2023; Hong et al. 2023). Recent prompt-based approaches like IC-DST (Hu et al. 2022), SERI-DST (Lee and Lee 2024), InstructTODS (Chung et al. 2023), and FNCTOD (Li et al. 2024) leverage in-context learning but still struggle to translate broad language understanding into precise, structured dialogue states—particularly in multi-domain scenarios.

Knowledge representation frameworks offer promising solutions (Pan et al. 2024), through either semantic parsing (Lan and Jiang 2020; Ye et al. 2021) or retrieval-augmented approaches (Lewis et al. 2020; Jiang et al. 2022), though each approach has limitations. While more sophisticated knowledge fusion techniques like RAG (Lewis et al. 2020) and KG prompting (Zhang et al. 2024) have demonstrated improved performance, they still face challenges with retrieval accuracy and computational efficiency (Mavromatis and Karypis 2024;

Dehghan et al. 2024). GNNs (Gori, Monfardini, and Scarselli 2005; Veličković et al. 2017; Mavromatis and Karypis 2025; Luo et al. 2025) show promise for modeling complex relational structures, yet their application to dialogue understanding remains underdeveloped. Existing graph-based DST methods (Chen et al. 2020) often rely on static, predefined ontologies that cannot dynamic dialogue evolution.

Computational efficiency presents another significant challenge, particularly as models grow more complex to handle sophisticated dialogue understanding tasks. MoE architectures (Shazeer et al. 2017; Xue et al. 2024) have demonstrated parameter efficiency in large-scale language models (Jiang et al. 2024), though their application to DST remains unexplored. Similarly, agent-based approaches like ReAct (Yao et al. 2023) have shown promise for enhancing reasoning capabilities through explicit stepwise decomposition (Wei et al. 2022; Khot et al. 2022), but require further investigation for structured DST.

These limitations highlight the need for a hybrid architecture that combines graph-based dialogue representation for structural understanding, efficient computational allocation through expert routing, and specialized value generation for accurate knowledge extraction. Our work addresses these challenges through the integration of GNN, MoE, and specialized state value generators within a unified dialogue state tracking framework.

Method

We present a hybrid architecture that combines Graph Neural Networks (GNNs) and sequence model for structural dialogue understanding, Mixture of Experts (MoE) for efficient computation allocation, and state value generators for accurate knowledge extraction.

Intents and Slots Detection

We use two approaches for intent and slot detection, (i) a GNN that captures structural relationships across dialogue turns, and (ii) a T5 encoder-decoder model that leverages sequential context for comprehensive language understanding.

Graph Neural Network Following the success of GNNs(Gori, Monfardini, and Scarselli 2005) in processing graph-structured data, we construct our dialogue understanding framework using the Graph Attention Network (GAT)(Veličković et al. 2017; Brody, Alon, and Yahav 2021) to effectively model the complex interactional dynamics presentation in multi-turn conversations. Our approach leverages GAT's attention mechanism specifically for joint slot filling and intent detection tasks in dialogue understanding. In this work, we represent dialogue as graph structure $G = (V, E)$, where nodes $v_i \in V$ contains the contextual embedding of utterances, with special tokens [USER] and [ASSISTANT] to distinguish speaker types. The edges E create directed temporal connections between consecutive utterances, enabling information flow for both intent classification and slot labeling tasks.

For both slot filling and intent detection, we compute node representations through L layers of graph attention. At layer l, the representation is updated as:

$$h_i^{(l+1)} = \sigma\left(\sum_{j\in\mathcal{N}_i} \alpha_{ij}^{l} W^{(l)} h_j^{(l)}\right), \tag{1}$$

where, $\mathcal{N}_i$ is the neighborhood of node i, $W^{(l)} \in \mathbb{R}^{F'\times F}$ is a learnable weight matrix, $\alpha_{ij}^{(l)}$ are attention coefficients computed as:

$$\alpha_{ij}^{(l)} = \frac{\exp\left(\text{LeakyReLU}\left(\vec{a}^T[W^{(l)}h_i^{(l)}|W^{(l)}h_j^{(l)}]\right)\right)}{\sum_{k\in\mathcal{N}_i}\exp\left(\text{LeakyReLU}\left(\vec{a}^T[W^{(l)}h_i^{(l)}|W^{(l)}h_k^{(l)}]\right)\right)}, \tag{2}$$

where $|$ represents concatenation and $\vec{a}$ is a learnable attention vector.

For both slot filling and intent detection tasks, we perform utterance-level multi-label classification using the final node representations. We employ K independent attention heads and combine their outputs:

$$h_i^{(L)} = \sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j\in\mathcal{N}i} \alpha_{ij}^{k,(L-1)} W_k^{(L-1)} h_j^{(L-1)}\right). \tag{3}$$

The multi-head attention mechanism helps the model attend to different aspects of the dialogue context, which is particularly beneficial for capturing the relationships between slots and intents across multiple turns. This approach allows our model to jointly learn slot and intent representations while accounting for the dialogue history and conversational dynamics.

T5 Encoder-Decoder Model While graph-based approaches excel at modeling structural relationships, transformer-based sequence models have demonstrated exceptional capabilities in natural language understanding tasks (Raffel et al. 2020). The T5 architecture models intent detection and slot filling as a sequence-to-sequence generation task, where the input is the dialogue context and the output is a structured representation of intents and slots. We finetune the T5 model using a standard sequence-to-sequence objective that maximizes the probability of generating the correct structured output given the dialogue context.

State Value Generator

After slot and intent detection, we extract slot values using two approaches that share a common dynamic few-shot example selection mechanism.

Dynamic Few-Shot Example Selection To enable effective information retrieval for few-shot learning, we create dense vector representations of dialogue contexts. Each training example is encoded using Cohere's embedding model (AWS Documentation 2023), represented as $\mathbf{z}_i = \text{Embed}(f_{\text{combine}}(D_{i-1}^{\text{sys}}, D^{\text{user}}i, S_i))$, where f_{combine} constructs a structured text representation combining the previous system response, current user utterance, and identified slots. For a given dialogue turn t, we retrieve the most relevant examples by computing semantic similarity:

$$E_t = \text{TopK}(\mathbf{e} \in \mathcal{E} : \cos(\mathbf{z}_e, \mathbf{z}_t) > \tau_{\text{sim}}), \quad (4)$$

where $\mathbf{z}_t$ represents the embedding of the current turn and τ_{sim} is a similarity threshold. These embeddings are indexed in ChromaDB for efficient similarity-based retrieval. Each retrieved example $e \in E_t$ includes its ground truth slot s_i, value v_j pairs $V_e = \{(s_j, v_j)\}_{j=1}^{m_e}$, providing demonstration instances for the subsequent value generation methods.

LLM-Based Generation We utilize pre-trained decoder-only LLMs for slot value extraction through few-shot learning. For turn t, we construct input using dynamically retrieved examples E_t along with dialogue history and identified slots to generate structured slot-value pairs:

$$\mathbf{x}_t = f_{\text{concat}}(\boldsymbol{\delta}_{\text{prefix}}, E_t, \boldsymbol{\delta}_{\text{history}}, D_{0:t-1}, \boldsymbol{\delta}_{\text{current}}, D_t, \boldsymbol{\delta}_{\text{slots}}, S_t), \quad (5)$$

$$V_t = \text{LLM}_\theta(\mathbf{x}_t) = \{(s_i, v_i)\}_{i=1}^{k}, \quad (6)$$

where S_t are identified slots from the GNN module, $\boldsymbol{\delta}_\cdot$ represent delimiter tokens, and the model generates structured JSON responses with slot-value pairs.

ReAct Agent-Based Generation The ReAct agent utilizes structured reasoning to extract slot values through explicit analysis and reasoning steps in a single turn. Using the same retrieved examples E_t as the LLM approach, the agent performs compositional reasoning functions:

$$V_t = \mathcal{A}_{\boldsymbol{\phi}}(E_t, D_{0:t}, S_t) = (f_{\text{analysis}} \circ f_{\text{reasoning}} \circ f_{\text{json}})(E_t, D_{0:t}, S_t), \quad (7)$$

where f_{analysis} analyzes dialogue context with retrieved examples, $f_{\text{reasoning}}$ provides explicit reasoning traces for identified values, and f_{json} generates structured JSON outputs. This approach enhances interpretability while maintaining the same extraction protocols as the LLM-based method.

Mixture of Experts

To speed up learning and improve generalization across different scenarios (Jacobs et al. 1991) proposed using several different expert networks instead of a single model. Mixture of expert models have been extensively studied since then, consistently demonstrating both performance improvements and cost reductions (Yao et al. 2009; Liu et al. 2024; Vats et al. 2024). In this work we consider using gating strategy for model selection. When designing an effective gating function, two key criteria must be prioritized: (1) accurate discernment of both input and expert characteristics to enable assignment of similar data to the same expert, and (2) even distribution of input data among experts to prevent model collapse and ensure efficient utilization of all experts. Our routing architecture leverages domain-weighted voting strategy to direct dialogue turns to the most appropriate expert model. The system employs a BERT-based multi-label domain classifier trained on conversational history to identify active domains for each dialogue turn. The voting scheme for our proposed method is presented in Algorithm 1, where in this algorithm H represents the dialogue history, U is the current utterance, Acc_{GNN} and Acc_{T5} represent slot accuracies for respective models.

Algorithm 1: Routing Mechanism

Require: H, U
Ensure: $M \in (\text{GNN}, \text{T5})$
$D \leftarrow \text{DomainClassifier}(H, U)$
$\text{votes}_{GNN} \leftarrow 0$, $\text{votes}_{T5} \leftarrow 0$
for each domain $d \in D$ **do**
  if $\text{Acc}_{GNN} > \text{Acc}_{T5}$ **then**
    $\text{votes}_{GNN} \leftarrow \text{votes}_{GNN} + 1$
  else
    $\text{votes}_{T5} \leftarrow \text{votes}_{T5} + 1$
  end if
end for
if $\text{votes}_{GNN} \geq \text{votes}_{T5}$ **then**
  return GNN
else
  return T5
end if

Experiments

We evaluate our method for dialogue state tracking task on the MultiWOZ 2.2 dataset. Detailed ablation studies examining the impact of different GNN configurations, embedding models, context window sizes and loss weighting are provided in their respective sections. Latency analysis across different model configurations is presented separately as well.

Dataset: We conduct experiments on MultiWOZ 2.2 (Budzianowski et al. 2018), a multi-domain task-oriented dialogue dataset containing 10,438 human-human conversations across 7 domains (restaurant, hotel, attraction, taxi, train, hospital, police). This dataset provides rich annotations for intent detection and slot filling tasks, with 13 intent types and 25 slot types across approximately 13.7 average turns per dialogue. We specifically choose MultiWOZ 2.2 as it offers consistent annotation quality for intent and slot labels throughout the corpus, while later versions (2.3-2.4) primarily address co-reference annotations and test set corrections that are orthogonal to our focus. The dataset's multi-turn nature (with dialogues extending up to 30+ turns) makes it particularly suitable for evaluating contextual understanding capabilities. Additionally, MultiWOZ 2.2 remains a widely-used version in the dialogue understanding literature, enabling direct comparison with existing baselines and SOTA methods.

Evaluation Metrics: We report Joint Goal Accuracy (JGA) and Joint Turn Accuracy (JTA) for the slot-value generation. For intent and slot classification we report turn-level accuracy.

Experimental setup

Our experimental framework utilizes the multi-expert architecture described in Section . For the GNN, we utilize BERT-base (Devlin et al. 2019) to encode dialog histories with speaker tokens, which then feeds into a three-layer GATv2 network with three parallel MLP decoders to simultaneously predict intent, domain, and slot values, jointly optimized using binary cross-entropy loss. We use QLoRA-

Model	Configuration	Intent Acc	Slot Acc
T5	Small	86.69	73.76
GNN	BERT$_{base}$ + GNN$_{base}$	86.84	71.07
	BGE $_{small}$ + GNN$_{base}$	87.00	69.71
	BERT $_{base}$ + GNN$_{large}$	86.63	71.55
	(BERT $_{base}$ + GNN$_{large}$)$_{opt}$	86.53	**74.54**

Table 1: Test set performance for intent classification and slot identification across different model configurations. The GNN rows show variations in embedding models (BERT$_{base}$, BGE$_{small}$), architecture (GNN$_{base}$, GNN$_{large}$), and other optimization approaches (opt) including residual connections, BCE loss weighting and dialogue context window. Results demonstrate the impact of different design choices on model performance.

tuned T5-Small (Raffel et al. 2020) to transform concatenated dialog histories into structured domain-intent-slot combinations through sequence-to-sequence generation. For slot value extraction, we applied two approaches: (1) a retrieval-augmented generation (RAG) system and (2) a ReAct agent-based for slot-value generation. For both we use Llama 3.1 8B Instruct (Grattafiori et al. 2024) and Claude 3.7 Sonnet (AWS Sonnet Documentation 2023) as the LLM model. All models were trained on NVIDIA A10G GPU with 24GB memory, 8 vCPUs, 32GB RAM.

Results

We evaluate our proposed framework on the MultiWOZ 2.2 benchmark dataset, analyzing performance across multiple dimensions: intent and slot detection accuracy, value generation quality with different LLM configurations, domain-specific routing behavior, and overall dialogue state tracking performance compared to SOTA baselines. All experiments were conducted on the standard test split to ensure fair comparison with existing methods.

Intent and Slot Detection: Table 1 provides a detailed analysis of our model's performance on intent classification and slot identification tasks across different architectural configurations. Our GNN architecture demonstrates strong performance on both intent classification (86.53%) and slot identification (74.54%) when using the optimized configurations. The finetuned T5-Small encoder-decoder model shows comparable intent and slot classification performance (86.69% and 73.76% respectively). Ablation studies are provided in Appendix.

Dialog State Tracking using LLM: We evaluate three LLM configurations: zero-shot (no examples), few-shot (with in-context examples), and ReAct (reasoning and acting framework) across two LLM families. The results for this experiment are illustrated in Table 2. The ReAct based method dramatically improves Llama 3.1 8B performance, achieving 26.74% JGA improvement over zero-shot prompting and representing the best end-to-end LLM results on the test set.

Domain-Aware Routing Mechanism: As was explained, we introduce a routing architecture that leverages domain-specific characteristics to direct dialogue turns to the most appropriate expert model. Table 3 presents the domain-specific routing performance across each domain. The results show an interesting pattern where the router predominantly selects GNN for attraction 91.31% and train 93.48% domains, while favoring T5 for hotel 89.99%, restaurant 85.82%, and taxi 87.27% domains. This routing strategy reflects the inherent structural differences between domains. Attraction and train domains typically involve more complex entity relationships and key components to extract, which benefits from GNN's graph-structured reasoning capabilities. These domains often contain interconnected information about locations, schedules, and facilities that can be effectively modeled as graphs. Conversely, hotel, restaurant, and taxi domains tend to involve more straightforward slot-value pairs where T5's sequential processing excels.

Model	Type	JGA	JTA
Llama 3.1 8B	Zero	10.33	80.65
Llama 3.1 8B	Few	32.62	89.77
Llama 3.1 8B	ReAct	**37.07**	93.36
Claude 3.7 S.	Zero	32.50	90.82
Claude 3.7 S.	Few	**38.43**	91.48
Claude 3.7 S.	ReAct	37.09	91.48

Table 2: End-to-end LLM performance on MultiWOZ 2.2

Metric	Attraction	Hotel	Restaurant	Taxi	Train
GNN Routing %	**91.31**	10.01	14.18	12.73	**93.48**
T5 Routing %	8.69	**89.99**	**85.82**	**87.27**	6.52
Router Accuracy	**70.21**	**56.05**	**71.64**	**61.62**	**78.47**

Table 3: MoE Performance by domain showing routing distribution and accuracy metrics for each expert model.

Dialog State Tracking using MoE: Table 4 presents our proposed methods against baselines on the MultiWOZ dataset. We compare our approach with several SOTA baselines including DS-DST, Seq2Seq-DU, and LUNA (all using BERT$_{base}$ with 110M parameters), SPACE-3 (UniLM with $\sim$340M parameters), SPLAT (using both Longformer$_{base}$ and Longformer$_{large}$), Diable (T5v1.1$_{base}$), D3ST (T5 variants), and TOATOD (T5$_{small}$ and T5$_{base}$), which achieved strong results on the MultiWOZ leaderboard. Among the traditional baselines, TOATOD$_{base}$ achieved the highest performance at 63.79% JGA with 220M parameters, while D3ST$_{xxl}$ achieved 58.70% JGA at the substantial cost of 11B parameters. In contrast, our single models (T5$_{small}$ and GNN) achieve considerably higher performance with dramatically lower parameter counts. The GNN-based models with Llama 3.1 8B and Claude 3.7 Sonnet as value generators achieve 62.66% and 64.34% JGA respectively, while maintaining JTA scores above 97.6% with only 271M parameters. Our T5$_{small}$ architecture achieves 63.39% JGA with Llama 3.1 8B and 64.66% JGA with Claude 3.7 Sonnet, both outperforming the previous SOTA by significant margins while using less than 1% of the parameters compared to D3ST$_{xxl}$ (70M vs 11B). These results demonstrate the effectiveness

Method	Model(s)			JGA	JTA
DS-DST	BERT_{base}			51.70	-
Seq2Seq-DU	BERT_{base}			54.40	-
LUNA	BERT_{base}			56.13	-
SPACE-3	UniLM			57.50	-
SPLAT	Longformer_{base}			56.60	-
	$\text{Longformer}_{large}$			57.40	-
Diable	T5v1.1_{base}			56.48	-
D3ST	T5_{base}			56.10	-
	T5_{large}			54.20	-
	T5_{xxl}			58.70	-
TOATOD	T5_{small}			61.92	-
	T5_{base}			63.79	-
Single Expert$_{(ours)}$	BERT_{base}	+ GNN	+ Llama 3.1 8B	62.66	97.66
	BERT_{base}	+ GNN	+ Claude 3.7 S.	64.34	97.74
		T5_{small}	+ Llama 3.1 8B	63.39	97.44
		T5_{small}	+ Claude 3.7 S.	64.66	97.50
GEM$_{(ours)}$	(BERT_{base}	+ GNN) / T5_{small}	+ Llama 3.1 8B	**63.57**	97.56
	(BERT_{base}	+ GNN) / T5_{small}	+ Claude 3.7 S.	**65.19**	97.65

Table 4: This table presents comparative results of our proposed GEM (Graph-Enhanced Mixture) framework against existing SOTA dialogue state tracking models. GEM achieves superior JGA of 65.19% by efficiently combining GNN and T5 architectures with ReAct agent integration.

	Val		Test	
Model	**Intent**	**Slot**	**Intent**	**Slot**
BERT	87.46	72.67	86.84	**71.07**
BGE-s	87.36	70.36	87.00	69.71
BGE-b	86.38	69.50	85.47	68.05
DistilUSE	82.68	51.71	81.99	52.17

Table 5: Embedding model comparison.

	Val		Test	
α:β:γ	**Intent**	**Slot**	**Intent**	**Slot**
1:1:1	86.34	73.58	86.10	72.25
1:0.5:1	87.09	74.80	86.63	72.99
1:0.5:2	86.85	75.86	86.53	**74.54**
1:0.5:3	86.64	76.09	86.39	74.39
1:0.5:5	86.49	75.20	86.19	73.86

Table 6: Loss weighting schemes.

of our method and the agentic approach proposed in Section . Our GEM framework combining GNN and T5 experts with ReAct agent-based value generation further improves performance across both LLM families, achieving 63.57% JGA with Llama 3.1 8B and 65.19% JGA with Claude 3.7 Sonnet, demonstrating effective routing between complementary expert architectures through our domain-weighted voting strategy described in Algorithm 1, which dynamically routes dialogue turns based on conversational characteristics.

Analysis and Discussion

Our experimental results demonstrate that specialized architectural components with dynamic routing mechanisms significantly outperform pure LLM approaches for dialogue state tracking, while maintaining computational efficiency through selective expert activation.

Performance Analysis: As shown in Table 2, the MoE approach significantly outperforms end-to-end LLM methods, achieving 65.19% JGA compared to 32.50% JGA for zero-shot Claude (a 32.69 percentage point improvement). Our domain-specific routing analysis reveals that GNN models excel in domains with complex relational structures (attractions and train domains), while T5 performs better in domains with straightforward slot-value extraction patterns (hotels, restaurants, and taxis). The choice of LLM for value generation (Table 4) shows modest but consistent impact, with Claude 3.7 Sonnet achieving 1.62 percentage points higher JGA than Llama 3.1 8B in the MoE configuration (65.19% vs 63.57%) , though the ReAct framework provides the most substantial gains for Llama 3.1 8B.

Computational Efficiency: Our MoE framework achieves SOTA performance while maintaining computational efficiency through selective expert activation, using either the lightweight T5-Small (70M parameters) or GNN architecture (271M parameters), dramatically fewer than previous approaches like D3ST-T5XXL (11B parameters). As presented in Table 4, while Claude 3.7 Sonnet achieves marginally better performance (65.19% vs. 63.57% JGA in MoE configuration), Llama 3.1 8B offers significant advantages in deployment latency and computational cost with only a 1.62 percentage point JGA trade-off, enabling practical deployment scenarios where cost-effective, low-latency inference is prioritized.

Ablation Studies

To systematically evaluate and optimize the GNN component of our architecture, we conducted comprehensive ablation studies examining how different GNN configurations contribute to slot detection performance. We focus on four critical aspects: (1) embedding model selection, (2) GNN architecture configuration, (3) dialogue context window size, and (4) loss weighting strategies. All experiments use the MultiWOZ 2.2 dataset with consistent evaluation protocols to ensure optimal GNN performance before integration into our broader MoE framework.

Impact of Embedding Models: We investigated different pre-trained language models for utterance encoding, as shown in Table 5. BERT-base consistently outperforms other models, particularly for slot accuracy. While BGE-small (Xiao et al. 2024) shows comparable intent detection, it underperforms on slot accuracy. DistilUSE-multilingual (Reimers and Gurevych 2019) performs significantly worse on both metrics. These results suggest BERT-base's representations better capture the semantic information required for DST.

GNN Architecture Configuration: We investigated how different GNN configurations affect performance using our BERT-base embedding approach with full dialogue context.

Configuration	Validation		Test	
	Intent	Slot	Intent	Slot
Small (1L, 4H, 128d)	86.76	70.98	86.67	69.38
Baseline (2L, 8H, 256d)	87.46	72.67	86.84	71.07
Large (3L, 12H, 512d)	87.67	73.11	86.63	71.55
XLarge (4L, 16H, 768d)	87.51	72.29	86.54	70.67
Small (1L, 4H, 128d) r.	87.50	71.71	86.75	69.70
Baseline (2L, 8H, 256d) r.	87.52	73.38	**87.03**	71.28
Large (3L, 12H, 512d) r.	86.66	73.88	86.75	72.15
XLarge (4L, 16H, 768d) r.	87.33	75.29	86.63	**73.33**

Table 7: Performance comparison of different GNN configurations. 'r.' denotes with residual connections.

Context Window	Validation		Test	
	Intent	Slot	Intent	Slot
0 (current turn only)	76.93	50.38	75.61	50.08
1 turn	84.50	64.70	83.79	63.85
2 turns	85.77	70.27	84.93	69.18
3 turns	86.13	71.79	85.09	70.43
5 turns	86.15	73.73	85.53	71.76
10 turns	86.34	73.58	86.10	**72.25**
Full history	86.66	73.88	86.75	72.15

Table 8: Performance comparison of different context window sizes.

We denote configurations as (Layers, Heads, Dimension). In configurations with residual connections (denoted as 'w. residual'), each GAT layer's input is added to its output, facilitating gradient flow and preserving information from earlier layers in deeper architectures. Table 7 presents results for three configurations of increasing complexity. The results suggest that intent detection benefits from the more compact Baseline configuration, while slot prediction accuracy increase as model size increasing. For our following experiments, we adopted the Large configuration that balances parameter efficiency and model performance, prioritizing slot prediction accuracy which has greater impact on the overall JGA.

Context Window Size: Using BERT-base encodings and GNN-Large configuration, we systematically varied the dialogue context window size to quantify its impact on performance. The empirical results shown in Table 8 demonstrate the critical importance of dialogue context for accurate state tracking. The most substantial improvements occur within the first two turns. This dramatic improvement confirms that DST fundamentally requires cross-turn reasoning.

Loss Weighting Strategies: We investigated different weighting schemes for the multi-task learning objective, using a normalized weighted sum formulation:

$$L_{total} = \frac{\alpha \times L_{intent} + \beta \times L_{domain} + \gamma \times L_{slot}}{\alpha + \beta + \gamma}, \quad (8)$$

where α, β, and γ represent the weights for intent detection, domain classification, and slot detection tasks respectively. Table 6 shows performance across different configurations where performance peaks with configuration (1:0.5:2) which achieves 74.54% slot accuracy on the test set. This confirms our hypothesis that slot prediction represents a more challenging task requiring additional optimization emphasis.

Model	Params	Latency (ms)
BERT_{base} + GNN	271M	5
T5_{small}	70M	21
GEM	341M	12

Table 9: Comparison of latency per turn for slot extraction models.

ReAct Agent LLM	Latency (ms)
Llama 3.1 8B	2050
Claude 3.7 Sonnet	3950

Table 10: Comparison of latency per turn for LLM-based value extraction. Llama 3.1 8B measurements on 4x NVIDIA L40S, Claude 3.7 Sonnet via Amazon Bedrock API

Latency Analysis

We analyze the computational efficiency of our proposed models by measuring inference latencies across different components of our system, as shown in Tables 9 and 10. For slot extraction efficiency, our individual expert models demonstrate highly optimized performance with BERT_{base} + GNN achieving 5ms per turn with 271M parameters, while T5_{small} provides 21ms per turn with only 70M parameters, representing different points on the efficiency-capacity trade-off curve for slot extraction tasks. Our GEM method successfully combines both expert architectures through an optimized gating mechanism, achieving a balanced 12ms per turn latency with 341M total parameters, demonstrating that the mixture approach provides a practical middle ground between the speed of BERT-based processing and the generative capabilities of T5 while maintaining deployment feasibility for real-time dialogue systems. Regarding LLM value generation trade-offs, we observe substantial differences in inference performance where Llama 3.1 8B deployed on 4x NVIDIA L40S GPUs achieves 2050ms per turn compared to Claude 3.7 Sonnet's 3950ms per turn, representing a 48% latency reduction that makes Llama 3.1 8B particularly attractive for latency-sensitive applications where the modest 1.62 percentage point decrease in Joint Goal Accuracy is an acceptable trade-off for nearly doubling the inference speed.

Conclusion and Future Work

In this work, we present GEM, a novel Mixture-of-Experts framework for Dialogue State Tracking that addresses the fundamental limitations of end-to-end LLM approaches in structured information extraction. By dynamically routing between a Graph Neural Network that captures dialogue structure and a T5 encoder-decoder for sequence modeling, followed by slot-value extraction using an LLM-based ReAct approach, our method achieves 65.19% Joint Goal Accuracy on MultiWOZ 2.2, outperforming SOTA methods like TOATOD (63.79%). Our analysis reveals that specialized architectures excel at capturing dialogue dynamics that general-purpose LLMs struggle with, particularly in multi-domain conversations requiring precise slot-value tracking.

Future work will extend the framework to cross-dataset evaluation, explore modeling the MoE router using additional dimensions such as dialogue text and chains of domains /

intents / slots from previous turns, and investigate knowledge distillation into smaller unified models for resource-constrained deployment. These directions advance more efficient and deployable dialogue understanding systems.

References

AWS Documentation. 2023. Embedding Models on Amazon Bedrock. https://docs.aws.amazon.com/bedrock/latest/userguide/model-parameters-embed.html. Accessed: 2024.

AWS Sonnet Documentation. 2023. Sonnet 3.7 on Amazon Bedrock. https://docs.aws.amazon.com/bedrock/latest/userguide/model-evaluation-type-judge-prompt-claude-sonnet37.html. Accessed: 2024.

Bang, N.; Lee, J.; and Koo, M.-W. 2023. Task-optimized adapters for an end-to-end task-oriented dialogue system. *arXiv preprint arXiv:2305.02468*.

Bang, Y.; Cahyawijaya, S.; Lee, N.; Dai, W.; Su, D.; Wilie, B.; Lovenia, H.; Ji, Z.; Yu, T.; Chung, W.; et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.

Bebensee, B.; and Lee, H. 2023. Span-Selective Linear Attention Transformers for Effective and Robust Schema-Guided Dialogue State Tracking. In Rogers, A.; Boyd-Graber, J.; and Okazaki, N., eds., *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 78–91. Toronto, Canada: Association for Computational Linguistics.

Brody, S.; Alon, U.; and Yahav, E. 2021. How attentive are graph attention networks? *arXiv preprint arXiv:2105.14491*.

Budzianowski, P.; Wen, T.-H.; Tseng, B.-H.; Casanueva, I.; Ultes, S.; Ramadan, O.; and Gašić, M. 2018. Multiwoz–a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *arXiv preprint arXiv:1810.00278*.

Chen, H.; Wang, J.; Wang, W.; and Xu, Y. 2025. Improving zero-shot chain-of-thought reasoning across languages with rectification and self-optimization prompting. *The Journal of Supercomputing*, 81(10): 1–21.

Chen, L.; Lv, B.; Wang, C.; Zhu, S.; Tan, B.; and Yu, K. 2020. Schema-guided multi-domain dialogue state tracking with graph attention neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 7521–7528.

Chung, W.; Cahyawijaya, S.; Wilie, B.; Lovenia, H.; and Fung, P. 2023. InstructTODS: Large language models for end-to-end task-oriented dialogue systems. *arXiv preprint arXiv:2310.08885*.

Dehghan, M.; Alomrani, M. A.; Bagga, S.; Alfonso-Hermelo, D.; Bibi, K.; Ghaddar, A.; Zhang, Y.; Li, X.; Hao, J.; Liu, Q.; et al. 2024. EWEK-QA: Enhanced web and efficient knowledge graph retrieval for citation-based question answering systems. *arXiv preprint arXiv:2406.10393*.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, 4171–4186.

Feng, Y.; Wang, Y.; and Li, H. 2021. A Sequence-to-Sequence Approach to Dialogue State Tracking. arXiv:2011.09553.

Gori, M.; Monfardini, G.; and Scarselli, F. 2005. A new model for learning in graph domains. In *Proceedings. 2005 IEEE international joint conference on neural networks, 2005.*, volume 2, 729–734. IEEE.

Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

He, W.; Dai, Y.; Yang, M.; Sun, J.; Huang, F.; Si, L.; and Li, Y. 2022. SPACE-3: Unified Dialog Model Pre-training for Task-Oriented Dialog Understanding and Generation. arXiv:2209.06664.

Heck, M.; Lubis, N.; Ruppik, B.; Vukovic, R.; Feng, S.; Geishauser, C.; Lin, H.-C.; van Niekerk, C.; and Gašić, M. 2023. ChatGPT for zero-shot dialogue state tracking: A solution or an opportunity? *arXiv preprint arXiv:2306.01386*.

Heck, M.; van Niekerk, C.; Lubis, N.; Geishauser, C.; Lin, H.-C.; Moresi, M.; and Gašić, M. 2020. Trippy: A triple copy strategy for value independent neural dialog state tracking. *arXiv preprint arXiv:2005.02877*.

Hong, R.; Zhang, H.; Zhao, H.; Yu, D.; and Zhang, C. 2023. Faithful question answering with monte-carlo planning. *arXiv preprint arXiv:2305.02556*.

Hu, Y.; Lee, C.-H.; Xie, T.; Yu, T.; Smith, N. A.; and Ostendorf, M. 2022. In-context learning for few-shot dialogue state tracking. *arXiv preprint arXiv:2203.08568*.

Huang, J.; and Chang, K. C.-C. 2022. Towards reasoning in large language models: A survey. *arXiv preprint arXiv:2212.10403*.

Hudeček, V.; and Dusek, O. 2023. Are Large Language Models All You Need for Task-Oriented Dialogue? In Stoyanchev, S.; Joty, S.; Schlangen, D.; Dusek, O.; Kennington, C.; and Alikhani, M., eds., *Proceedings of the 24th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, 216–228. Prague, Czechia: Association for Computational Linguistics.

Jacobs, R. A.; Jordan, M. I.; Nowlan, S. J.; and Hinton, G. E. 1991. Adaptive mixtures of local experts. *Neural computation*, 3(1): 79–87.

Jiang, A. Q.; Sablayrolles, A.; Roux, A.; Mensch, A.; Savary, B.; Bamford, C.; Chaplot, D. S.; Casas, D. d. l.; Hanna, E. B.; Bressand, F.; et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.

Jiang, J.; Zhou, K.; Zhao, W. X.; and Wen, J.-R. 2022. Unikgqa: Unified retrieval and reasoning for solving multi-hop question answering over knowledge graph. *arXiv preprint arXiv:2212.00959*.

Khot, T.; Trivedi, H.; Finlayson, M.; Fu, Y.; Richardson, K.; Clark, P.; and Sabharwal, A. 2022. Decomposed prompting: A modular approach for solving complex tasks. *arXiv preprint arXiv:2210.02406*.

Lan, Y.; and Jiang, J. 2020. Query graph generation for answering multi-hop complex questions from knowledge bases. Association for Computational Linguistics.

Lee, J.; and Lee, G. G. 2024. Inference is All You Need: Self Example Retriever for Cross-domain Dialogue State Tracking with ChatGPT. *arXiv preprint arXiv:2409.06243*.

Lesci, P.; Fujinuma, Y.; Hardalov, M.; Shang, C.; and Marquez, L. 2023. Diable: Efficient Dialogue State Tracking as Operations on Tables. In *Findings of the Association for Computational Linguistics: ACL 2023*, 9697–9719. Association for Computational Linguistics.

Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474.

Li, Z.; Chen, Z. Z.; Ross, M.; Huber, P.; Moon, S.; Lin, Z.; Dong, X. L.; Sagar, A.; Yan, X.; and Crook, P. A. 2024. Large language models as zero-shot dialogue state tracker through function calling. *arXiv preprint arXiv:2402.10466*.

Liu, J.; Tang, P.; Wang, W.; Ren, Y.; Hou, X.; Heng, P.-A.; Guo, M.; and Li, C. 2024. A survey on inference optimization techniques for mixture of experts models. *arXiv preprint arXiv:2412.14219*.

Luo, L.; Zhao, Z.; Haffari, G.; Phung, D.; Gong, C.; and Pan, S. 2025. GFM-RAG: graph foundation model for retrieval augmented generation. *arXiv preprint arXiv:2502.01113*.

Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*, 1(3): 3.

Mavromatis, C.; and Karypis, G. 2024. Gnn-rag: Graph neural retrieval for large language model reasoning. *arXiv preprint arXiv:2405.20139*.

Mavromatis, C.; and Karypis, G. 2025. GNN-RAG: Graph Neural Retrieval for Efficient Large Language Model Reasoning on Knowledge Graphs. In *Findings of the Association for Computational Linguistics: ACL 2025*, 16682–16699.

Pan, S.; Luo, L.; Wang, Y.; Chen, C.; Wang, J.; and Wu, X. 2024. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 36(7): 3580–3599.

Raffel, C.; Shazeer, N.; Roberts, A.; Lee, K.; Narang, S.; Matena, M.; Zhou, Y.; Li, W.; and Liu, P. J. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140): 1–67.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Shazeer, N.; Mirhoseini, A.; Maziarz, K.; Davis, A.; Le, Q.; Hinton, G.; and Dean, J. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. *arXiv preprint arXiv:1701.06538*.

Vats, A.; Raja, R.; Jain, V.; and Chadha, A. 2024. The evolution of mixture of experts: A survey from basics to breakthroughs.

Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; and Bengio, Y. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Wang, K.; Duan, F.; Wang, S.; Li, P.; Xian, Y.; Yin, C.; Rong, W.; and Xiong, Z. 2023. Knowledge-driven cot: Exploring faithful reasoning in llms for knowledge-intensive question answering. *arXiv preprint arXiv:2308.13259*.

Wang, Y.; Zhao, J.; Bao, J.; Duan, C.; Wu, Y.; and He, X. 2022. LUNA: Learning Slot-Turn Alignment for Dialogue State Tracking. In Carpuat, M.; de Marneffe, M.-C.; and Meza Ruiz, I. V., eds., *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 3319–3328. Seattle, United States: Association for Computational Linguistics.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Wu, C.-S.; Madotto, A.; Hosseini-Asl, E.; Xiong, C.; Socher, R.; and Fung, P. 2019. Transferable multi-domain state generator for task-oriented dialogue systems. *arXiv preprint arXiv:1905.08743*.

Xiao, S.; Liu, Z.; Zhang, P.; Muennighoff, N.; Lian, D.; and Nie, J.-Y. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*, 641–649.

Xue, F.; Zheng, Z.; Fu, Y.; Ni, J.; Zheng, Z.; Zhou, W.; and You, Y. 2024. Openmoe: An early effort on open mixture-of-experts language models. *arXiv preprint arXiv:2402.01739*.

Yao, B.; Walther, D.; Beck, D.; and Fei-Fei, L. 2009. Hierarchical mixture of classification experts uncovers interactions between brain regions. *Advances in Neural Information Processing Systems*, 22.

Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; and Cao, Y. 2023. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*.

Ye, X.; Yavuz, S.; Hashimoto, K.; Zhou, Y.; and Xiong, C. 2021. RNG-KBQA: Generation augmented iterative ranking for knowledge base question answering. *arXiv preprint arXiv:2109.08678*.

Zhang, J.; Hashimoto, K.; Wu, C.-S.; Wang, Y.; Yu, P.; Socher, R.; and Xiong, C. 2019. Find or classify? dual strategy for slot-value predictions on multi-domain dialog state tracking. In *Proceedings of the 9th International Workshop on Spoken Dialogue Systems Technology*, 154–167.

Zhang, Q.; Dong, J.; Chen, H.; Zha, D.; Yu, Z.; and Huang, X. 2024. Knowgpt: Knowledge graph based prompting for large language models. *Advances in Neural Information Processing Systems*, 37: 6052–6080.

Zhao, J.; Gupta, R.; Cao, Y.; Yu, D.; Wang, M.; Lee, H.; Rastogi, A.; Shafran, I.; and Wu, Y. 2022. Description-driven task-oriented dialog modeling. *arXiv preprint arXiv:2201.08904*.