Event-Guided Slot Interaction for Multi-Domain Dialogue State Tracking

Ying Xia and Wei Liu*
School of Computer Engineering and Science
Shanghai University, China
{shineshadow_xy, liuw}@shu.edu.cn

Abstract

Multi-domain Dialogue State Tracking (DST) requires discourse coherence that transcends independent slot-filling. Most existing approaches rely on statistical regularities within static schemas, failing to capture the semantic coordination governing simultaneous slot updates. In this paper, we propose Event-DST, which models latent events as cognitive organizing units to dynamically coordinate slot interactions. By projecting dialogue context into a continuous semantic space, our model induces a dynamic structural bias to enforce pragmatic consistency. This structural guidance is integrated via a dual-stream fusion strategy that balances top-down structural constraints with bottom-up textual precision. Experimental results on two benchmarks demonstrate the superiority of our framework, providing an interpretable and parameter-efficient path toward robust dialogue understanding.

1 Introduction

Task-Oriented Dialogue (TOD) systems aim to assist users in accomplishing specific goals, such as restaurant reservation and hotel booking. As a core component of TOD, Dialogue State Tracking (DST) extracts user goals and constraints at each turn, maintaining a structured belief state to support policy learning and response generation (Wen et al., 2017).

Most existing DST approaches predict each slot independently from contextual representations (Wu et al., 2019; Heck et al., 2020), without explicitly modeling coordination among slot updates. While effective for isolated turn-level predictions, this independence assumption becomes problematic in multi-domain dialogues, where multiple slots are often updated within the same semantic event (Kim et al., 2021). In such cases, independently predicting operation decisions (e.g., UPDATE and CARRYOVER) may result in partial updates, where some slots are correctly revised while others incorrectly retain previous values. These locally inconsistent belief states can further introduce erroneous constraints that propagate across subsequent turns, as illustrated in Figure 1(a).

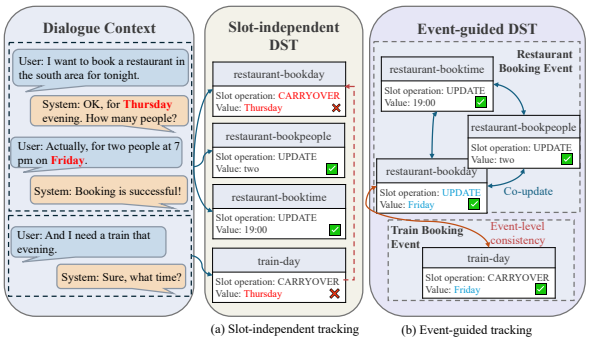

Figure 1: Comparison of tracking paradigms. (a) Slot-independent prediction may produce uncoordinated slot operations and cross-event inconsistencies. (b) Event-guided modeling uses latent event semantics to coordinate slot operations.

To map inter-slot dependencies, recent work incorporates structured modeling techniques, including schema graphs (Chen et al., 2020) and graph neural networks (Lin et al., 2021; Zhang et al., 2022). While Large Language Models (LLMs) have introduced sophisticated state induction through function calling (Li et al., 2024) and noetic graph reasoning (Li et al., 2025), these methods still rely on statistically induced relations, often failing to distinguish incidental co-updates from coherent semantic transitions (Ye et al., 2021; Liu et al., 2022). Crucially, existing frameworks lack a unified semantic organizing unit to dynamically coordinate slot interactions as dialogue goals evolve (Chen et al., 2020; Li et al., 2025).

In this work, we argue that dialogue is not merely a sequence of slot-filling operations but a coherent discourse governed by underlying semantic events.

*Corresponding author.

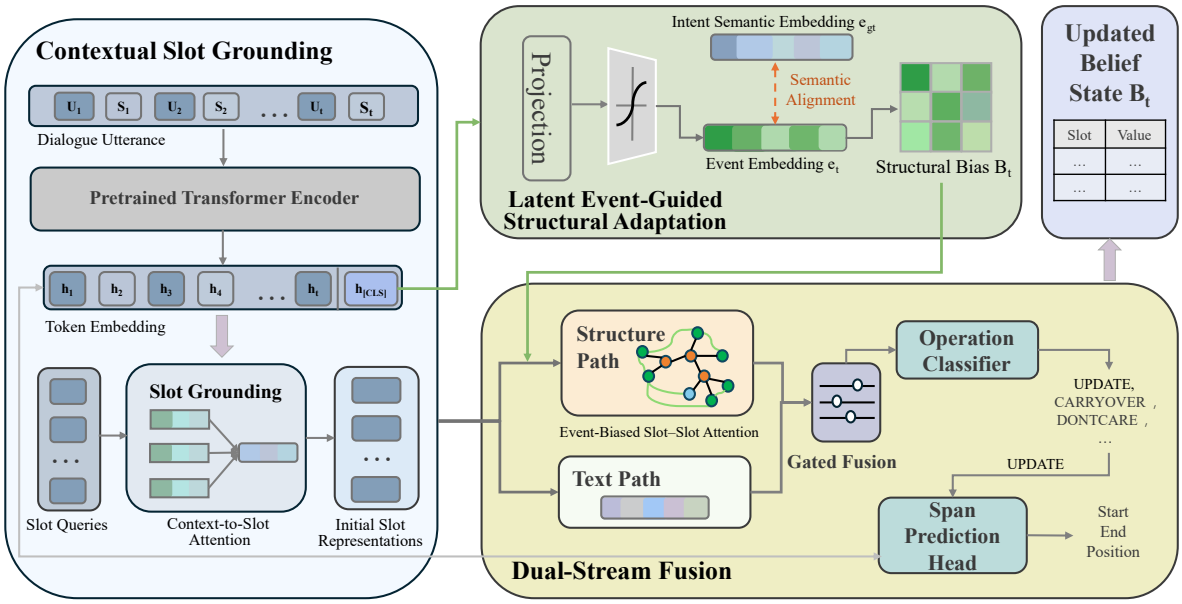

Figure 2: The overall architecture of our Event-DST model.

We propose that high-level event semantics provide the missing link between unstructured context and structured slot consistency. We define an "event" not merely as a discrete user intent, but as a latent semantic condition that dictates which slots should interact and how. By grounding slot interactions in these latent events, we can dynamically modulate the information flow between slots, prioritizing connections that are relevant to the current dialogue state. As conceptually illustrated in Figure 1(b), modeling latent events as unified semantic conditions enables event-guided co-updates–ensuring that multiple slots, such as *bookday* and *bookpeople*, are updated in unison. This mechanism effectively eliminates the partial updates and local inconsistencies that frequently plague slot-independent paradigms.

Building on this insight, we present a novel framework: *Event-guided Dialogue State Tracker* (Event-DST). Unlike previous methods that rely on static graphs, our approach introduces a dynamic *event-guided structural bias*. Specifically, we project dialogue contexts into a continuous event space and use these representations to generate an adaptive bias term, which modulates the attention mechanism between slots. Furthermore, to balance the benefits of raw contextual understanding and structured reasoning, we design a *dual-stream fusion mechanism*. This mechanism processes slot representations through two parallel paths—a text-driven path for precise value extraction and a structure-enhanced path for consistency modeling—before adaptively fusing them via a gated integration layer.

To summarize, our contributions are three-fold: (1) We highlight the role of dialogue-level event semantics in coordinating slot updates for dialogue state tracking. (2) We propose an event-guided DST framework that dynamically regulates inter-slot interactions by integrating contextual slot representations with event-conditioned structural dependencies. (3) Experimental results on multi-domain benchmarks validate the effectiveness of the proposed approach.

2 Related Work

2.1 Semantic Organizing Units

Mainstream Dialogue State Tracking (DST) paradigms have largely adopted a slot-centric formulation, treating each slot as an independent prediction unit. Early representative approaches, such as SUMBT (Lee et al., 2019) and BERT-DST (Chao and Lane, 2019), rely on non-parametric slot–utterance matching or discriminative classification to estimate values directly from the dialogue context. Although this decomposition scales well to multi-domain settings, it ignores the semantic connections between slots. Consequently, these models often fail to maintain consistency when multiple slots must be updated simultaneously.

To address the decision complexity of direct value prediction, subsequent research introduced

slot-level operations as intermediate decision units. For instance, TRADE (Wu et al., 2019) employs slot gates to determine update types before generation, while TripPy (Heck et al., 2020) characterizes state transitions via copy mechanisms from dialogue history or memory. Although these formulations improve efficiency by filtering unchanged slots, they remain fundamentally slot-centric: they model *how* individual slots change but lack a unified semantic rationale for *why* specific subsets of slots (e.g., *hotel-area* and *hotel-price*) are updated jointly.

Beyond local slot modeling, prior research has explored higher-level semantic abstractions to capture global dialogue dynamics. TreeDST (Cheng et al., 2020) imposes hierarchical constraints among dialog acts and slots, whereas latent variable models (Min et al., 2020) introduce hidden states to approximate implicit dialogue modes. Other works incorporate discourse-level structures to regularize long-horizon state evolution (Ouyang et al., 2020; Li et al., 2024). Nevertheless, these methods often depend on rigid schemas or weakly structured latent representations, which limits their adaptability to open-ended conversational shifts. To overcome these constraints, a semantic organizing unit is required to provide the rationale for joint updates. By transitioning from independent slot-filling to event-driven inference, the model ensures that co-occurring updates remain logically consistent with the broader discourse goal.

2.2 Structured Slot Modeling

Complementary to the semantic unit of update is the explicit modeling of structural dependencies between slots. Prior graph-enhanced DST studies have modeled domain-slot and inter-slot relations through structured graph representations (Zhou and Small, 2019; Lin et al., 2021). Within this line, schema-aware approaches, such as Schema-Guided DST (Chen et al., 2020) and CSFN (Zhang et al., 2022), inject prior domain knowledge by constructing static schema graphs. In these frameworks, slots and domains function as nodes within a fixed topology, employing Graph Attention Networks (GATs) or similar propagation mechanisms to facilitate information flow across semantically related slots. Although pre-computed graphs mitigate data sparsity, their reliance on invariant adjacency matrices presumes constant slot correlations. This rigidity restricts the model's ability to adapt to context-dependent shifts in active domains.

To overcome this limitation, subsequent studies have pivoted toward dynamic structure induction, allowing slot dependencies to evolve alongside the dialogue context. STAR (Ye et al., 2021) leverages self-attention to learn soft slot dependencies, while DSGFNet (Liu et al., 2022) explicitly predicts evolving graph structures conditioned on the dialogue context. Hybrid approaches like NSR-Graph (Li et al., 2025) further extend this by integrating static schema priors with dynamically updated relations. Despite the increased flexibility of these dynamic mechanisms, the induced structures are largely derived from statistical regularities in the data. Consequently, they capture observable correlations—identifying that slots often change together—but fail to model the underlying semantic coordination driven by the specific event being executed. In contrast, anchoring structural induction in latent event semantics provides the guidance necessary to filter spurious correlations common in statistical modeling. Such a mechanism ensures that inter-slot interactions are governed by precise semantic constraints rather than surface-level co-occurrence regularities.

3 Methodology

Our design is motivated by the linguistic observation that surface-level utterances are often ambiguous (Kawamoto, 2026), while the underlying communicative goal remains relatively stable. Although task-oriented dialogues are often associated with coarse domain-intent labels, dialogue state transitions can be more effectively modeled through higher-level semantic conditions that coordinate related slot updates across turns. Building on this view, we operationalize an event as a continuous semantic representation inferred from the dialogue context. At each turn t, this event representation captures the current communicative condition that determines which slots should interact and how strongly they should influence each other. Rather than relying on rigid, pre-defined schemas that impose invariant slot relations, our event representation induces soft, turn-specific dependencies between slots and adapts slot interactions to the current dialogue phase.

Based on this theoretical grounding, we present Event-DST (as shown in Figure 2), a framework that realizes event-guided slot interaction through three integrated modules: (1) a context encoder that grounds dialogue history into initial slot rep-

resentations (§3.2); (2) a latent event-guided interaction module that induces dynamic structural constraints to modulate slot-level information flow (§3.3); and (3) a dual-stream fusion decoder that balances global structural coherence with local textual precision (§3.4).

3.1 Problem Formulation

We formulate Multi-Domain Dialogue State Tracking (DST) not merely as a static classification task, but as a dynamic inference process of recovering the user's belief state from the surface discourse. Formally, let the surface discourse at turn t be denoted as $\mathcal{H}_t = [(U_1, S_1), \ldots, (U_t, S_t)]$, where U and S denote user and system utterances, respectively. The objective is to infer the belief state $\mathcal{B}_t = \{(s_i, v_{t,i})\}_{i=1}^N$ based on $\mathcal{H}_t$, where $\mathcal{S} = \{s_i\}_{i=1}^N$ represents the set of predefined slots. To model the logical transitions inherent in this process, we decompose the state update into operation prediction and value span extraction. For each slot, the model first predicts an operation label, such as CARRYOVER, UPDATE, or DONTCARE. Span extraction is applied only to slots whose operation is UPDATE, while non-update slots either inherit previous values or take predefined special values.

3.2 Contextual Slot Grounding

To bridge the gap between continuous linguistic signals and discrete slot values, the Contextual Slot Grounding module maps the raw discourse $\mathcal{H}_t$ into a semantic feature space. We employ a pre-trained encoder ($\text{BERT}_{\text{base}}$ (Devlin et al., 2019)) to map the input $\mathcal{H}_t$ into a sequence of contextual embeddings $\mathbf{H}_t = [\mathbf{h}_{\text{CLS}}, \mathbf{h}_1, \ldots, \mathbf{h}_L] \in \mathbb{R}^{(L+1)\times d}$. Here, $\mathbf{h}_{\text{CLS}}$ serves as the global context anchor, summarizing the overall intent of the current turn.

Conditioned on these contextual embeddings, we derive slot-specific representations through a learnable query-driven attention mechanism. Each slot s_i is associated with a learnable query $\mathbf{q}_i$, which is used to retrieve slot-relevant evidence from the token-level representations. Specifically, we employ scaled dot-product attention (Vaswani et al., 2017) to compute the relevance between slot s_i and contextual token $\mathbf{h}_j$:

$$a_{ij}^{\text{ctx}} = \frac{(\mathbf{W}_Q \mathbf{q}_i)^\top (\mathbf{W}_K \mathbf{h}_j)}{\sqrt{d_k}}. \tag{1}$$

The attention weights are normalized over all contextual tokens:

$$\alpha_{ij} = \frac{\exp(a_{ij}^{\text{ctx}})}{\sum_{\ell=1}^{L} \exp(a_{i\ell}^{\text{ctx}})}. \tag{2}$$

The initial slot representation is then obtained by aggregating token-level evidence:

$$\mathbf{r}_i^{(0)} = \sum_{j=1}^{L} \alpha_{ij} \mathbf{h}_j. \tag{3}$$

This yields context-aware slot representations $\{\mathbf{r}_i^{(0)}\}_{i=1}^N$, which capture slot-specific textual evidence but do not yet model inter-slot dependencies.

3.3 Latent Event-Guided Structural Adaptation

As discussed above, slot dependencies in multi-domain dialogue are not fixed across turns, but vary with the current communicative phase. We formalize an event at turn t as a continuous semantic representation $\mathbf{e}_t = f_\theta(\mathcal{H}_t)$, inferred from the dialogue history. This event representation serves as the semantic condition for generating a turn-specific structural bias over slots. In this way, slot dependencies are allowed to evolve adaptively according to the inferred event semantics rather than being imposed by a static schema graph.

3.3.1 Semantic Event Inference

To instantiate f_θ, we project the surface discourse into a latent semantic space anchored by natural language descriptions derived from domain-intent metadata. These descriptions summarize the communicative goal and the slots typically involved in that goal, such as "*book a taxi by pickup location, destination, departure time, and arrival time*". They are used as semantic anchors for shaping the event space during training, rather than as fixed slot-slot adjacency structures.

Given the global context representation $\mathbf{h}_{\text{CLS}}$, the model first derives an intermediate latent representation $\mathbf{z}_t$:

$$\mathbf{z}_t = \phi\left(\mathbf{W}_{\text{event}}^{(z)} \mathbf{h}_{\text{CLS}} + \mathbf{b}_{\text{event}}\right), \tag{4}$$

where $\phi(\cdot) = \tanh(\cdot)$ constrains $\mathbf{z}_t \in \mathbb{R}^K$ to $(-1, 1)$, facilitating stable optimization. The intermediate representation is then projected into the event semantic space via $\mathbf{e}_t = \mathbf{W}_{\text{proj}} \mathbf{z}_t$.

During training, $\mathbf{e}_t$ is aligned with a frozen semantic prototype derived from the corresponding

event description, as described in §3.5. During inference, $\mathbf{e}_t$ is produced solely from the dialogue context and no gold event description or intent label is required.

3.3.2 Dynamic Structural Bias Injection

Given the inferred event representation $\mathbf{e}_t$, we generate a *Dynamic Structural Bias* $\mathbf{B}_t$ to modulate slot-to-slot attention. Instead of learning a static adjacency matrix, the bias matrix is generated on-the-fly:

$$\mathbf{B}_t = \text{reshape}(\mathbf{W}_b \mathbf{e}_t) \in \mathbb{R}^{N \times N}. \quad (5)$$

This matrix $\mathbf{B}_t$ represents the *event-conditioned interaction strength* between slots at turn t. We incorporate this bias into the self-attention mechanism among slots:

$$a_{ij}^{\text{slot}} = \underbrace{\frac{(\mathbf{W}_Q^s \mathbf{r}_i^{(0)})^\top (\mathbf{W}_K^s \mathbf{r}_j^{(0)})}{\sqrt{d_k}}}_{\text{Content Interaction}} + \underbrace{\mathbf{B}_{t,ij}}_{\text{Event-Guided Bias}}. \quad (6)$$

By adding $\mathbf{B}_{t,ij}$, the model explicitly promotes information flow between semantically entangled slots (e.g., *Train-Dest* and *Train-Time*) while suppressing irrelevant connections, effectively performing soft graph induction. The structure-enhanced representation is then computed as:

$$\mathbf{r}_i^{\text{struct}} = \sum_{j=1}^{N} \text{softmax}_j(a_{ij}^{\text{slot}}) \mathbf{W}_V^s \mathbf{r}_j^{(0)}. \quad (7)$$

3.4 Dual-Stream Fusion and Prediction

Human language comprehension relies on the interplay between bottom-up signal processing and top-down cognitive priors. Reflecting this cognitive synergy, we propose a Dual-Stream Fusion strategy to synthesize precise local context with global structural consistency. To this end, we maintain two parallel processing pathways:

1. **Text-Driven Path ($\mathbf{r}_i^{\text{text}}$)**: This path preserves slot-specific textual evidence for value extraction. It is computed by applying a lightweight linear projection to the initial slot representation:

$$\mathbf{r}_i^{\text{text}} = f_{\text{text}}(\mathbf{r}_i^{(0)}) = \mathbf{W}_{\text{text}} \mathbf{r}_i^{(0)} + \mathbf{b}_{\text{text}}. \quad (8)$$

2. **Structure-Enhanced Path ($\mathbf{r}_i^{\text{struct}}$)**: Output of the event-guided interaction, focusing on maintaining cross-slot consistency.

These streams are adaptively fused via a learnable gate $\mathbf{g}_i$:

$$\mathbf{g}_i = \sigma(\mathbf{W}_g [\mathbf{r}_i^{\text{text}}; \mathbf{r}_i^{\text{struct}}] + \mathbf{b}_g), \quad (9)$$

$$\mathbf{r}_i^{\text{hybrid}} = \mathbf{g}_i \odot \mathbf{r}_i^{\text{struct}} + (1 - \mathbf{g}_i) \odot \mathbf{r}_i^{\text{text}}. \quad (10)$$

This gating mechanism allows the model to dynamically rely on structural priors when the text is ambiguous, or trust the text when the value is explicit.

Finally, the operation class is predicted via $\mathbf{o}_i = \mathbf{W}_{\text{cls}} \mathbf{r}_i^{\text{hybrid}}$ over a set of predefined actions, where $\mathbf{o}_i$ denotes the operation logits for slot s_i. For UPDATE, the new slot value is extracted using text-driven features $[\mathbf{r}_i^{\text{text}}; \mathbf{h}_j]$ to preserve local context precision. For CARRYOVER, the slot inherits the value $v_{t-1,i}$ from the previous turn. For special operations such as DONTCARE, the corresponding values are assigned directly.

3.5 Optimization with Semantic Supervision

The model is trained end-to-end using a weighted multi-task objective that combines the primary state tracking task with an auxiliary semantic alignment task:

$$\mathcal{L} = \mathcal{L}_{\text{dst}} + \lambda_{\text{align}} \mathcal{L}_{\text{align}}, \quad (11)$$

where λ_{align} is a hyperparameter balancing the tasks.

3.5.1 Dialogue State Tracking Loss

The DST objective encompasses both operation classification and span extraction:

$$\mathcal{L}_{\text{dst}} = \sum_{i=1}^{N} \left(\lambda_{\text{cls}} \mathcal{L}_{\text{cls}}^{(i)} + \mathbb{I}[c_{t,i} = \text{UPDATE}] \mathcal{L}_{\text{span}}^{(i)} \right), \quad (12)$$

where $c_{t,i} \in \mathcal{C}$ denotes the gold operation label for slot s_i at turn t, and $\mathcal{C}$ is the predefined operation set introduced in the prediction module. $\mathcal{L}_{\text{cls}}^{(i)}$ is the cross-entropy loss for operation prediction, and $\mathcal{L}_{\text{span}}^{(i)}$ sums the cross-entropy losses for start and end positions.

3.5.2 Auxiliary Semantic Alignment Loss

To semantically ground the inferred event representation, we introduce an auxiliary alignment objective. Let D_{event} denote the natural language description associated with the active domain-intent event at turn t. For MultiWOZ, these descriptions are constructed from the existing ontology and domain-intent metadata without introducing additional turn-level annotations. During training,

active events are automatically inferred from the existing ontology and turn-level state changes, without requiring additional manual event annotations. For example, *Find-Restaurant* is mapped to "find a restaurant by considering food type, price range, area, and name", while *Book-Restaurant* is mapped to "book a restaurant with day, time, and number of people". When multiple domain-intent events are active in the same turn, we average their corresponding description embeddings.

We encode D_{event} with a frozen BERT encoder to obtain the semantic prototype $\mathbf{e}_{\text{gt}}$. Both $\mathbf{e}_t$ and $\mathbf{e}_{\text{gt}}$ are represented in the same event semantic space $\mathbb{R}^{d_e}$. We align the projected event representation $\mathbf{e}_t$, rather than the intermediate representation $\mathbf{z}_t$, with this semantic prototype:

$$\mathcal{L}_{\text{align}} = 1 - \frac{\mathbf{e}_t^\top \mathbf{e}_{\text{gt}}}{\|\mathbf{e}_t\| \|\mathbf{e}_{\text{gt}}\|}. \tag{13}$$

The lower bound of $\mathcal{L}_{\text{align}}$ is 0, achieved when $\mathbf{e}_t$ and $\mathbf{e}_{\text{gt}}$ have cosine similarity 1. This objective regularizes the event space so that the generated structural bias $\mathbf{B}_t$ is grounded in semantic prototypes rather than purely statistical co-occurrence patterns.

4 Experiments

4.1 Experimental Settings

4.1.1 Datasets

Experiments are conducted on MultiWOZ 2.2 (Zang et al., 2020) and MultiWOZ 2.4 (Ye et al., 2022). Both datasets follow the standard split of 8,438 training, 1,000 validation, and 1,000 test dialogues across five domains. We use MultiWOZ 2.2 as the main benchmark for its rectified labels and span annotations. Additionally, we evaluate on MultiWOZ 2.4, which cleans over 41% of the test data but keeps the training set noisy, allowing us to test model robustness against imperfect supervision.

4.1.2 Baselines

We compare our framework against representative baselines across multiple DST paradigms. Specifically, we include sequence generation and hybrid copy-based approaches, such as TRADE (Wu et al., 2019), SimpleTOD (Hosseini-Asl et al., 2020), Seq2Seq-DU (Feng et al., 2021), AG-DST (Tian et al., 2021), TripPy (Heck et al., 2020), DS-DST (Zhang et al., 2020), SOM-DST (Kim et al., 2020), and ECDG-DST (Zhu and Xu, 2025). In addition, we consider large-scale pre-training and prompting-based methods, including UniLM (Dong et al., 2019), SPACE-3 (He et al., 2022), D3ST (Zhao et al., 2022), and SDP-DST (Lee et al., 2021). Furthermore, to benchmark explicit dependency modeling, we incorporate structure-aware approaches that capture slot relations via matching or graph-based reasoning, namely SUMBT (Lee et al., 2019) and NSR-Graph (Li et al., 2025). Together, these baselines cover a comprehensive spectrum of paradigms, ranging from traditional discriminative architectures and explicitly structured graphs to large-scale generative models. This diversity enables a rigorous assessment of the effectiveness of our event-guided structural modeling.

Table 1: JGA (%) comparison on MultiWOZ 2.2 and MultiWOZ 2.4. For Event-DST, we report the average results of five runs with different random seeds. ± denotes standard deviation. Best scores are in bold.

Models	**MW 2.2**	**MW 2.4**
TRADE (Wu et al., 2019)	45.40	55.05
SimpleTOD (Hosseini-Asl et al., 2020)	56.45	–
SUMBT (Lee et al., 2019)	49.70	61.86
DS-DST (Zhang et al., 2020)	51.70	–
SOM-DST (Kim et al., 2020)	–	66.78
ECDG-DST (Zhu and Xu, 2025)	51.40	–
TripPy (Heck et al., 2020)	53.50	64.75
Seq2Seq-DU (Feng et al., 2021)	54.40	67.10
AG-DST (Tian et al., 2021)	57.26	–
UniLM (Dong et al., 2019)	54.25	–
SPACE-3 (He et al., 2022)	57.50	–
D3ST (Zhao et al., 2022)	54.20	70.80
SDP-DST (Lee et al., 2021)	57.60	–
NSR-Graph (Li et al., 2025)	52.90	–
Event-DST (Ours)	**57.76** (±0.10)	**71.60** (±0.13)

4.1.3 Evaluation

We adopt Joint Goal Accuracy (JGA) as the primary metric. JGA considers a dialogue turn correct only if the predicted values for all tracked slots exactly match the ground truth. This strict turn-level criterion ensures that any single slot error penalizes the entire prediction, serving as a rigorous measure of global consistency.

4.1.4 Implementation Details

The model is implemented in PyTorch and trained on a single NVIDIA GeForce RTX 4090D GPU (24GB). We employ *BERT-base-uncased* as the context encoder. For the auxiliary alignment task, semantic targets $\mathbf{e}_{\text{gt}}$ are derived from natural lan-

guage intent descriptions. We use manual mappings for MultiWOZ (e.g., `Find-Restaurant` → "*find a restaurant by considering food type, price range, area, and name*") and encode them offline using a frozen BERT instance. The resulting prototype embeddings have dimension $d_e = 768$, which matches the dimension of the projected event representation $\mathbf{e}_t$ produced by $\mathbf{W}_{\text{proj}}$.

Hyperparameters are tuned via grid search based on the JGA of the development set. For the coefficients in the training objective (Eq. 11), the auxiliary alignment weight λ_{align} and the operation classification weight λ_{cls} are set to 0.02 and 0.5, respectively. The model is optimized using Adam with an initial learning rate of 1×10^{-4}, $\epsilon = 10^{-8}$, and a dropout rate of 0.3.

4.2 Main Results

Table 1 presents the performance of our model and the baselines. Overall, Event-DST achieves the best average JGA among the compared models on both benchmarks, with 57.76% on MultiWOZ 2.2 and 71.60% on MultiWOZ 2.4. On MultiWOZ 2.2, Event-DST improves over the strongest baseline SDP-DST (Lee et al., 2021) by 0.16%. On MultiWOZ 2.4, Event-DST improves over the strongest baseline D3ST (Zhao et al., 2022) by 0.80%. Although the improvement on MultiWOZ 2.2 is modest, the consistent performance across datasets and the reported standard deviations provide empirical support for the effectiveness and stability of Event-DST. Based on the comparisons in Table 1, we make the following observations.

(1) **Event-DST improves over slot-centric methods.** Compared with methods that primarily perform slot-level prediction, Event-DST obtains better results. On MultiWOZ 2.2, it outperforms TripPy (Heck et al., 2020) and DS-DST (Zhang et al., 2020) by 4.26% and 6.06%, respectively. On MultiWOZ 2.4, it also surpasses SOM-DST (Kim et al., 2020) and TripPy (Heck et al., 2020) by 4.82% and 6.85%, respectively. These results suggest that modeling dependencies among slots is useful for improving state tracking beyond isolated slot-level decisions.

(2) **Dynamic adaptation outperforms invariant structural priors.** Event-DST outperforms structure-aware baselines on MultiWOZ 2.2, improving over SUMBT (Lee et al., 2019) and NSR-Graph (Li et al., 2025) by 8.06% and 4.86%, respectively. Unlike methods that rely on fixed schema-level relations, Event-DST generates event-conditioned structural biases on the fly, allowing slot interactions to adapt to the current dialogue context. This suggests that dynamic, context-dependent slot modeling is more effective than invariant structural priors.

Table 2: Ablation results on MultiWOZ 2.2 and 2.4. "w/o Semantic Alignment" removes the auxiliary event-semantic alignment objective, while "w/o Structure" removes the entire event-guided structural adaptation module.

	MW 2.2		MW 2.4	
Model	**JGA**	**Δ**	**JGA**	**Δ**
Full Model	**57.76**	–	**71.60**	–
w/o Semantic Alignment	56.52	↓1.24	70.45	↓1.15
w/o Structure	54.10	↓3.66	69.57	↓2.03

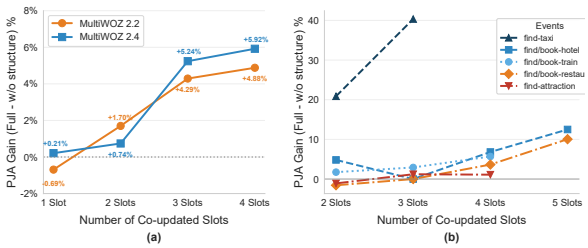

Figure 3: (a) Partial Joint Accuracy (PJA) gains of the full model over the w/o Structure baseline under increasing co-update complexity on MultiWOZ 2.2 and MultiWOZ 2.4. (b) Event-level PJA gains of the full model over the w/o Structure baseline under increasing co-update complexity (MultiWOZ 2.2).

4.3 Ablation Study and Analysis

To thoroughly assess the contribution of each component in Event-DST, we conduct ablation studies on the MultiWOZ 2.2 and MultiWOZ 2.4 test sets. All variants share the same encoder and training configuration.

4.3.1 Impact of Structural Components

We compare the full model against two variants to verify the efficacy of our proposed mechanisms:

- **w/o Semantic Alignment**: Removes the auxiliary alignment loss ($\lambda_{\text{align}} = 0$) and the event semantic targets. The latent event z_t is learned in an unsupervised manner without explicit semantic guidance.

- **w/o Structure**: Removes the entire event-guided structural adaptation module. Slots are predicted independently based on the dialogue context, serving as a text-only baseline.

As shown in Table 2, both components contribute to the final performance. Removing Semantic Alignment results in a drop of 1.24% on MultiWOZ 2.2 and 1.15% on MultiWOZ 2.4, suggesting that learning latent events solely from the DST objective provides limited guidance for structuring the event space. Introducing auxiliary semantic alignment provides additional supervision that helps regularize the event space and stabilize the induced structural biases. In contrast, removing the Structure module leads to the largest performance degradation, with drops of 3.66% on MultiWOZ 2.2 and 2.03% on MultiWOZ 2.4, confirming the importance of event-guided inter-slot dependency modeling over independent slot prediction.

4.3.2 Analysis of Structural Efficacy

We analyze how the event-guided structure improves tracking in complex scenarios involving simultaneous slot updates and cross-turn consistency.

Intra-Event Coordination We evaluate the effect of event-guided structural modeling on turns requiring coordinated slot updates. A turn is considered a *co-update turn* if two or more slots are updated simultaneously. Performance is measured using Partial Joint Accuracy (PJA), which requires all slot operations to be predicted correctly.

As shown in Figure 3(a), the advantage of event-guided modeling increases with co-update complexity. For single-slot updates, the performance difference between the full model and the *w/o Structure* baseline is marginal. In contrast, for multi-slot updates, the full model consistently outperforms the baseline, with the gap widening as more slots are updated within the same turn. This trend holds across both MultiWOZ 2.2 and 2.4, indicating that event-guided structure is particularly beneficial when multiple interdependent slots must be coordinated.

Figure 3(b) further analyzes this effect at the event level. Events involving tightly coupled constraints, such as `find-taxi`, exhibit the largest gains under event-guided modeling, especially when three or more related slots are updated simultaneously. This indicates that the latent event representation effectively recovers the underlying relational structure of the dialogue (e.g., that a taxi's destination is fundamentally dependent on the active restaurant slot), which surface-level statistical models often treat as incidental co-occurrences.

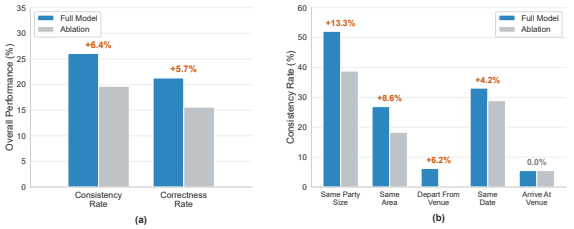

Figure 4: Analysis of Semantic Consistency on MultiWOZ 2.2. (a) Comparison of overall consistency rates between the full model and the w/o structure baseline. (b) Detailed breakdown of consistency improvement by specific semantic relation types (e.g., Party Size, Area).

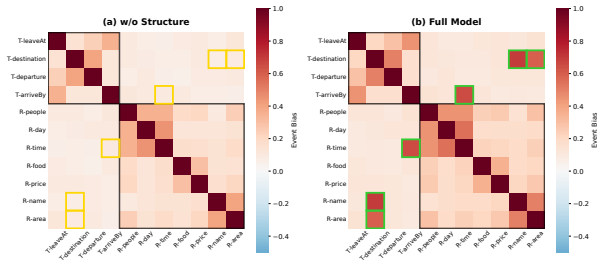

Figure 5: **Visualization of slot interaction patterns for the utterance "*I need a taxi to the restaurant*". (a) w/o Structure:** The model isolates domains, failing to capture dependencies between *Taxi* and *Restaurant*. **(b) Full Model:** The event mechanism activates cross-domain attention, enabling the model to reference *Restaurant* slots for *Taxi* updates. **Note:** Prefixes 'T-' and 'R-' denote *Taxi* and *Restaurant* domains.

Cross-Event Semantic Consistency. Beyond turn-level accuracy, we assess the model's ability to maintain global discourse consistency—checking if slots governed by shared constraints (e.g., `SAME_AREA`) maintain identical values. Figure 4 shows that the Full Model improves the overall consistency rate by +6.4% absolute over the baseline. Critically, the gain reaches +13.3% for party size constraints, suggesting that the event mechanism functions as a global semantic anchor. It propagates pragmatic constraints across disjoint turns, ensuring that the belief state remains coherent even as the conversational focus shifts across different semantic phases.

Overall, these findings demonstrate that event-guided structural modeling enhances coordinated slot updates within individual events and improves semantic consistency across related slots throughout the dialogue, jointly accounting for the observed gains in overall JGA.

4.4 Case Study

Figure 5 visualizes how the proposed framework Event-DST coordinates semantic information across domains for the request: *"I need a taxi to the restaurant"*. This scenario highlights cross-domain coreference, where the `Taxi-Destination` is elliptically referenced from an entity established in the restaurant domain. As shown in Figure 5(a), the w/o Structure variant fails to bridge domain boundaries, leading to a state update failure. In contrast, the Full Model (Figure 5(b)) detects the latent "Restaurant-to-Taxi" event and generates a context-dependent structural bias. This enables pragmatic anchoring, allowing the model to leverage `Restaurant` attributes to resolve the update. By grounding these interactions in a unified representation, Event-DST achieves structural reasoning aligned with the coherent nature of human dialogue.

5 Conclusion

In this paper, we propose Event-DST, a framework that reformulates dialogue tracking as an event-guided discourse rather than a sequence of independent operations. By modeling latent events as cognitive units, our approach induces a dynamic structural bias to coordinate slot interactions as conversational goals evolve. This is complemented by a dual-stream strategy that synthesizes top-down pragmatic constraints with bottom-up signals, ensuring global belief consistency without sacrificing local extraction precision.

Empirical results demonstrate that explicit relational modeling can improve DST performance within a parameter-efficient encoder-based framework. The induced structural biases provide transparent interpretability for complex cross-domain transitions. By bridging neural learning with linguistic priors, Event-DST establishes a robust, linguistically plausible path for modeling communicative intent in task-oriented dialogue.

Limitations

This work has several limitations. First, Event-DST is instantiated with BERT-base as the context encoder, and semantic prototypes are obtained using a frozen BERT encoder. Although the event-guided structural bias operates on slot representations and is conceptually independent of the encoder backbone, we have not systematically evaluated its generality across alternative pretrained models such as RoBERTa, DeBERTa, T5-small, or BERT-large.

Second, Event-DST does not yet integrate recent LLM-based DST paradigms, such as prompting, function calling, or structured generation. Our current framework focuses on parameter-efficient encoder-based modeling with explicit event-guided structural bias. Future work could explore how latent event representations can be incorporated into LLM-based state tracking to guide tool use, constrain structured generation, and provide interpretable intermediate reasoning signals.

References

Guan-Lin Chao and Ian Lane. 2019. Bert-dst: Scalable end-to-end dialogue state tracking with bidirectional encoder representations from transformer. *arXiv preprint arXiv:1907.03040*.

Lu Chen, Boer Lv, Chi Wang, Su Zhu, Bowen Tan, and Kai Yu. 2020. Schema-guided multi-domain dialogue state tracking with graph attention neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7521–7528.

Jianpeng Cheng, Devang Agrawal, Héctor Martínez Alonso, Shruti Bhargava, Joris Driesen, Federico Flego, Dain Kaplan, Dimitri Kartsaklis, Lin Li, Dhivya Piraviperumal, and 1 others. 2020. Conversational semantic parsing for dialog state tracking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8107–8117.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pages 4171–4186.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. *Advances in neural information processing systems*, 32.

Yue Feng, Yang Wang, and Hang Li. 2021. A sequence-to-sequence approach to dialogue state tracking. In *Proceedings of the 59th annual meeting of the association for computational linguistics and the 11th international joint conference on natural language processing (volume 1: long papers)*, pages 1714–1725.

Wanwei He, Yinpei Dai, Min Yang, Jian Sun, Fei Huang, Luo Si, and Yongbin Li. 2022. Unified dialog model pre-training for task-oriented dialog understanding and generation. In *Proceedings of the 45th International ACM SIGIR Conference on Research and*

Development in Information Retrieval, pages 187–200.

Sebastian Heck, Carel van Niekerk, Nurul Lubis, Christian Geishauser, Hsien-Chin Lin, Marco Moresi, and Milica Gasic. 2020. Trippy: A triple copy strategy for value independent neural dialogue state tracking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. *Advances in Neural Information Processing Systems*, 33:20179–20191.

Satoshi Kawamoto. 2026. Interactive field: A descriptive study of shared states in extended human–ai interaction.

Sungdong Kim, Sohee Kim, Dong-Hun Lee, and Young-Bum Kim. 2021. Performance impact of slot relation modeling in dialogue state tracking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*.

Sungdong Kim, Sohee Yang, Gyuwan Kim, and Sang-Woo Lee. 2020. Efficient dialogue state tracking by selectively overwriting memory. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 567–582.

Chia-Hsuan Lee, Hao Cheng, and Mari Ostendorf. 2021. Dialogue state tracking with a language model using schema-driven prompting. *arXiv preprint arXiv:2109.07506*.

Hwaran Lee, Jinsik Lee, and Tae-Yoon Kim. 2019. Sumbt: Slot-utterance matching for universal and scalable belief tracking. *arXiv preprint arXiv:1907.07421*.

Jingyang Li, Shengli Song, Sitong Yan, Guangneng Hu, Chengen Lai, and Yulong Zhou. 2025. Advanced dialog state tracking with noetic graphs for complex human-machine interactions. *Pattern Recognition*, page 111842.

Zekun Li, Zhiyu Chen, Mike Ross, Patrick Huber, Seungwhan Moon, Zhaojiang Lin, Xin Luna Dong, Adithya Sagar, Xifeng Yan, and Paul A Crook. 2024. Large language models as zero-shot dialogue state tracker through function calling. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8688–8704.

Weizhe Lin, Bo-Hsiang Tseng, and Bill Byrne. 2021. Knowledge-aware graph-enhanced gpt-2 for dialogue state tracking. *arXiv preprint arXiv:2104.04466*.

Yifan Liu, Zheng Zhang, and Yu Wang. 2022. Dsgfnet: Dynamic schema graph fusion network for dialogue state tracking. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.

Qingkai Min, Libo Qin, Zhiyang Teng, Xiao Liu, and Yue Zhang. 2020. Dialogue state induction using neural latent variable models. *arXiv preprint arXiv:2008.05666*.

Yawen Ouyang, Moxin Chen, Xinyu Dai, Yinggong Zhao, Shujian Huang, and Jiajun Chen. 2020. Dialogue state tracking with explicit slot connection modeling. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pages 34–40.

Xin Tian, Liankai Huang, Yingzhan Lin, Siqi Bao, Huang He, Yunyi Yang, Hua Wu, Fan Wang, and Shuqi Sun. 2021. Amendable generation for dialogue state tracking. *arXiv preprint arXiv:2110.15659*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Tsung-Hsien Wen, David Vandyke, Nikola Mrksic, Milica Gasic, Lina M. Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. Neural belief tracker: Data-driven dialogue state tracking. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*.

Chien-Sheng Wu, Andrea Madotto, Ehsan Hosseini-Asl, Caiming Xiong, Richard Socher, and Pascale Fung. 2019. Transferable multi-domain state generator for task-oriented dialogue systems. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*.

Fanghua Ye, Jarana Manotumruksa, and Emine Yilmaz. 2022. Multiwoz 2.4: A multi-domain task-oriented dialogue dataset with essential annotation corrections to improve state tracking evaluation. In *Proceedings of the 23rd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 351–360.

Fanghua Ye, Jarana Manotumruksa, Qiang Zhang, Shenghui Li, and Emine Yilmaz. 2021. Slot self-attentive dialogue state tracking. In *Proceedings of the web conference 2021*, pages 1598–1608.

Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang, and Jindong Chen. 2020. Multiwoz 2.2: A dialogue dataset with additional annotation corrections and state tracking baselines. *arXiv preprint arXiv:2007.12720*.

Jianguo Zhang, Kazuma Hashimoto, Chien-Sheng Wu, Yao Wang, Philip S Yu, Richard Socher, and Caiming Xiong. 2020. Find or classify? dual strategy for slot-value predictions on multi-domain dialog state tracking. In *Proceedings of the ninth joint conference on lexical and computational semantics*, pages 154–167.

Zheng Zhang, Yu Wang, and Qian Zhu. 2022. Efficient context and schema fusion networks for multi-domain dialogue state tracking. In *Proceedings of*

the 60th Annual Meeting of the Association for Computational Linguistics.

Jeffrey Zhao, Raghav Gupta, Yuan Cao, Dian Yu, Mingqiu Wang, Harrison Lee, Abhinav Rastogi, Izhak Shafran, and Yonghui Wu. 2022. Description-driven task-oriented dialog modeling. *arXiv preprint arXiv:2201.08904*.

Li Zhou and Kevin Small. 2019. Multi-domain dialogue state tracking as dynamic knowledge graph enhanced question answering. *arXiv preprint arXiv:1911.06192*.

Meng Zhu and Xiaolong Xu. 2025. Ecdg-dst: A dialogue state tracking model based on efficient context and domain guidance for smart dialogue systems. *Computer Speech & Language*, 90:101741.