Workflow-Guided Response Generation for Task-Oriented Dialogue

Anonymous ACL submission

Abstract

Task-oriented dialogue (TOD) systems aim to achieve specific goals through interactive dialogue. Such tasks usually involve following specific workflows, i.e. executing a sequence of actions in a particular order. While prior work has focused on supervised learning methods to condition on past actions, they do not explicitly optimize for compliance to a desired workflow. In this paper, we propose a novel framework based on reinforcement learning (RL) to generate dialogue responses that are aligned with a given workflow. Our framework consists of ComplianceReward, a metric designed to evaluate how well a generated response executes the specified action, combined with an RL optimization process that utilizes an interactive sampling technique. We evaluate our approach on two TOD datasets, Action-Based Conversations Dataset (ABCD) (Chen et al., 2021a) and MultiWOZ 2.2 (Zang et al., 2020) on a range of automated and human evaluation metrics. Our findings indicate that our RL-based framework outperforms baselines and is effective at generating responses that both comply with the intended workflows while being expressed naturally and fluently.

1 Introduction

Task-oriented dialogue (TOD) focuses on creating conversational systems that assist users in attaining specific objectives. While prior TOD literature has extensively looked at predicting user intents and identifying relevant slots and values (Henderson et al., 2014; Wei et al., 2018; Budzianowski et al., 2018; Byrne et al., 2019; Rastogi et al., 2020; Shalyminov et al., 2020; Balaraman et al., 2021), real-world interactions often involve nuanced workflows and optimizing for such workflows remains underexplored (Chen et al., 2021a; Hattami et al., 2022; Raimondo et al., 2023). Consider a customer support interaction where agents must follow multi-step procedures that adhere to company policies.

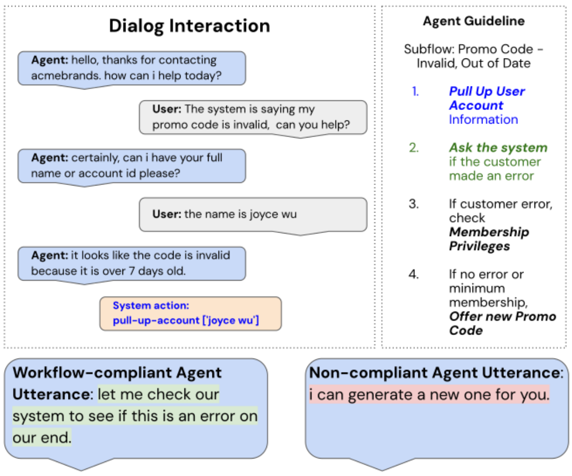

Figure 1: In this interaction, the customer requests assistance with an expired promo code. The agent must help the customer while following the steps in the agent guideline, consisting of a sequence of actions to be taken to resolve the issue. For example, offering to generate a new promo code without querying the system results in a non-workflow-compliant behavior.

For example, in Figure 1, a customer asks for help with an expired promotional code. A model that accounts for the user intent might respond reactively, offering to generate a new promo code. However, assisting the customer involves not only modeling their intent but also staying consistent with a workflow, in this case, the company policy. This involves the agent executing the necessary actions in the right order, such as pulling up account information and querying the system to make sure the customer qualifies for the promotion.

Many prior approaches in task-oriented dialogue (TOD), such as SimpleTOD (Hosseini-Asl et al., 2020) and PPTOD (Su et al., 2022), have employed supervised learning alongside utterance-level user intents and system dialogue acts (DAs) for system response generation. However, these frameworks lack explicit optimization for compliance in response generation, resulting in responses that may fail to execute the specified action. This problem arises because the response generators neither receive rewards nor penalties based on adherence to the specified actions. Additionally, there is a notable absence of a metric or model to quantitatively assess the degree of compliance, hindering the evaluation and training of response generators.

In this work, we tackle the problem of workflow-compliant response generation in TOD and propose an RL-based approach that addresses the limitations of existing systems. Our approach (COMPLIANCEOPT) employs RL with compliance scoring to construct training data for the Quark (Lu et al., 2022) framework. We evaluate our approach using the Action-Based Conversations Dataset (ABCD) (Chen et al., 2021a), a TOD dataset enriched with policy-based agent behavior constraints in the form of action sequences, and MultiWOZ 2.2 (Zang et al., 2020). Our experiments show that models integrating workflow information surpass baseline models, producing responses that adhere to policies while maintaining a natural and fluent tone. Furthermore, we observe that direct compliance optimization through RL can lead to additional enhancements in the workflow compliance levels of the dialogue system. We validate our results through automated metrics and human evaluations. Our contributions include:

- A reinforcement learning (RL)-based framework for training workflow-compliant response generators, based on an interactive sampling technique to optimize model behavior over multiple dialogue exchanges.[1]
- A new compliance metric based on a reward model validated against human evaluations.
- Evaluation on both automated and human evaluation metrics showing that our models, enhanced with workflow information and direct compliance optimization through RL, consistently outperform baselines.

2 Related Work

Task-oriented Dialogue. Recently, there has been an increase in TOD tasks and datasets (Budzianowski et al., 2018; Byrne et al., 2019; Wei et al., 2018; Rastogi et al., 2020), indicating a growing emphasis on advancing natural language processing techniques for practical applications. These datasets encompass diverse domains and enable researchers to tackle a wide spectrum of real-world challenges. However, previous benchmarks have predominantly focused on evaluating only some aspects of TOD systems, such as intent recognition and slot filling, with limited focus on aspects like workflow compliance (Chen et al., 2021a).

[1]We open-source our code at `github.com/ANON`.

Workflow Compliance. The problem of workflow compliance is closely related to, but distinct from, dialogue policy management. The primary objective of dialogue policy management is to predict the optimal dialogue action based on the current conversation state (Takanobu et al., 2019; He et al., 2022). In this context, dialogue actions represent intentions or decisions that are isolated to a single user query, such as "book a flight" or "find a nearby restaurant." In contrast, workflow compliance adopts a more holistic approach, considering the sequential workflow from the larger context of the conversation to define success. For example, offering a new promo code is only valid after a system check has been executed first (Figure 1). It emphasizes the fact that user interactions are not isolated actions but rather part of a continuous process with multiple steps. Raimondo et al. (2023) expands upon Chen et al. (2021a)'s work to show that models augmented with workflow-specific information such as workflow names or action plans can boost the generalizability of action prediction models, but does not consider the problem of generating workflow-compliant responses, which is a focus of our work.

SimpleTOD (Hosseini-Asl et al., 2020) is similar to the baselines in our work as both methods involve training an end-to-end model with interleaved actions and utterances as inputs. PPTOD (Su et al., 2022) also uses interleaved actions, but they use a more complex training pipeline that involves multi-task pretraining. While this can help improve performance, such baselines optimize for an objective that is different from our goal of increasing the compliance quality of generated responses.

Reinforcement Learning. RL has been successfully used to improve TOD systems (Pietquin et al., 2011; Gašić et al., 2013; Fatemi et al., 2016; Lewis et al., 2017; Singh et al., 2002). One application is training dialogue managers that maintain dialogue state transitions (Rieser and Lemon, 2011). Another is to use RL in conjunction with supervised learning to improve the quality of language generation, such as in (Lewis et al., 2017). This line of research applies similar techniques used in RL for general-domain dialogue generation, such as interleaving supervised learning and RL, offline and online RL, policy gradients, and Q-learning (Li et al., 2016b; Jang et al., 2022; Snell et al., 2023; Sodhi et al., 2023). Our work adopts a similar strategy of supervised learning followed by RL but introduces an interactive sampling step.

3 Problem Formulation

3.1 Workflow-Compliant Response Generation as an MDP

We formalize the problem of workflow-compliant response generation as a Markov Decision Process (MDP). Given a dataset of context-response pairs $\{x^i, y^i\}_{i=1}^N$, where context x is the conversation history, and response $y = \{y_1, \ldots, y_T\}$ is a target sequence of tokens.

Additionally, each dialogue is associated with a domain d representing the task (e.g., troubleshoot-site, subscription-inquiry). Every domain has an associated set of workflow G_d, which is a natural language description of the steps the system must follow to assist the user, as well as a sequence of actions W_d, which represents a flat action list or workflow sequence based on the guidelines. Fully compliant dialogues do not necessarily follow the full sequence W_d and may instead only include a subset of these actions since the guideline includes conditional branching. For example, in Figure 1, the step 3 is dependent on the result of step 2.

Each data instance, denoted as (x, y, G_d), can be viewed as an episode within an MDP, which we define as follows:

- **States,** $s_t \in \mathcal{S}$ is the context x, workflow G_d, and the partially generated sequence of tokens up to and including time step t, which we denote as $\hat{y}_{<t} := \hat{y}_1, \ldots, \hat{y}_t$.
- **Actions,** $a_t \in \mathcal{A}$ are the set of possible next tokens $\hat{y}_{t+1}$ from the vocabulary V.
- **Transition function,** $\mathcal{T}(s_{t+1}|s_t, a_t)$ is deterministic , as each state-action pair $(\hat{y}_{<t}, \hat{y}_{t+1})$ leads to a unique state $\hat{y}_{<t+1}$ for the next step.
- **Rewards,** $r_t : \mathcal{S} \times \mathcal{A} \rightarrow [0, 1]$ provide a measure of how well the generated response $\hat{y}$ executes the provided workflow G_d. It is a terminal reward. Since workflow compliance can be computed only after multiple exchanges, the reward is computed using *block evaluation*.
- **Horizon,** T represents the time span of each episode, concluding either when the current time step t exceeds T or when an end-of-sentence (EOS) token is generated.

The goal is to learn a policy $\pi : s_t \rightarrow a_t$ maximizing *return*, *i.e.* the cumulative reward over an episode $\mathbb{E}_\pi \sum_{t=0}^{T} \gamma^t r_t$. We assume undiscounted cumulative rewards, *i.e.* $\gamma = 1$.

Block Evaluation. One of our key observations is that compliance is not easily captured in a single dialogue response. For example, in a customer service use case, an agent may need to verify the identity of the user before proceeding to issue resolution. To successfully comply with the next workflow action e.g. `verify-identity`, the agent needs to take several steps. To better model and leverage this insight, we consider "blocks" of user and agent utterances when evaluating and optimizing for compliance. Blocks refer to the sequence of user and system utterances that occur between two action executions. We define an interaction "block" b as a list of user and system utterances between consecutive action executions by the system.

4 Approach

We introduce ComplianceOpt, which directly optimizes compliance with the specified workflow. We define compliance as the extent to which the generated system utterances adhere to the prescribed workflow action at turn t.

Algorithm 1 shows our overall training procedure, which is adapted from the Quark algorithm (Lu et al., 2022). The Quark framework is similar to Decision Transformer in that it treats RL as a sequence modeling problem (Chen et al., 2021b). After interactively sampled (Figure 2-(i)) generations are scored (Figure 2-(ii)), the rewards are quantized to produce reward tokens r_k, which are then used to condition the generations during training (Figure 2-(iii)).

4.1 Interactive Sampling

Diverging from the Quark method, we implement an interactive sampling step, using two distinct models, a system model, and a user simulator. This is because achieving workflow compliance often requires multiple dialogue turns between the participants. Consider a customer service agent who needs to gather a user's name, email, and order id to validate their purchase. This is a multi-turn process where the system needs to gather information over multiple dialogue turns of questions and answers.

We warm-start with a system model trained with standard autoregressive training. The user simula-

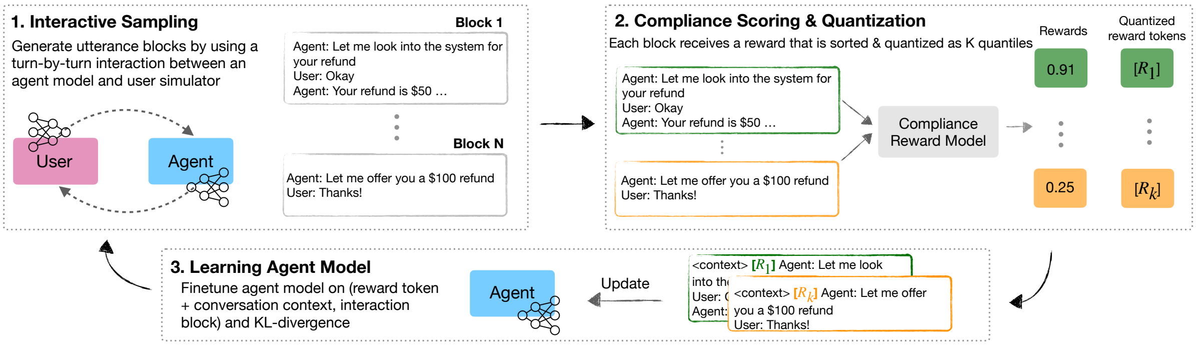

Figure 2: **Approach Overview.** RL optimizes the model towards better workflow compliance. Interaction-score pairs are processed into RL data in the Quark framework.

tor remains fixed during Quark training and is only used for the interactive sampling procedure. Given a dialogue context $c_0 = [u_0]$, the system model first samples an utterance, which is then concatenated to c_0, forming $c_1 = [u_0, s_1]$. Then, conditioned on c_1, the user simulator samples a user turn, forming another context $c_2 = [u_0, s_1, u_2]$. This process is repeated M times, which is a hyperparameter. We denote the generated user and system utterances *block* as b, which are fed alongside planned workflow actions as inputs to the ComplianceReward.

Our interactive sampling technique is independent of Quark and can be used as a sampling approach for other RL methods, such as proximal policy optimization (PPO) (Schulman et al., 2017). Previous studies, such as those by Zhang et al. (2022), have employed similar multi-utterance sampling techniques to simulate dialogue interactions. However, our approach diverges significantly from these methods. Instead of using a database of dialogue logs between real users and bots to construct RL data, our framework fully simulates responses for both users and systems, recognizing that completing an action typically requires multiple interactions. This introduces a more complex challenge for reinforcement learning (RL) optimization, as it necessitates the inclusion of simulated user responses within the block of utterances.

4.2 Compliance Scoring Model

To quantify compliance and use it as a reward for RL, we developed the ComplianceReward, which measures the alignment between the generated system utterances and the prescribed workflow action.

Reward modeling. We train the ComplianceReward using the reward modeling loss for ranking two responses (Ouyang et al., 2022).

$$l(\theta) = -\sum_{(p,b_w,b_l)\sim D} \log\left(\sigma\left(r_\theta(p,b_w) - r_\theta(p,b_l)\right)\right) \quad (1)$$

Algorithm 1 ComplianceOpt RL Training

Input: Initial Policy l_0, User Simulator μ, Dialogue Contexts C, reward $r(\cdot)$, KL weight β, number of quantiles K, number of interactions M, number of train iterations N

```
Make a copy l_θ of initial policy l_0.
for iteration = 1, 2, ··· , N do
    for c_i ∈ C do
        Do interactive_sample(l_θ, μ, M, c_i) to obtain b_i.          ▷ Interactive Sampling
        Add c_i, s_i, r(c_i, s_i) into data pool D                     ▷ Scoring
    end for
    D̄_i ← quantize
    for step = 1, 2, ··· , M do
        Draw a batch of data (c_i, b_i, r_{k_i}) from quantized data pool D̄_i
        Compute the objective in Eqn 2 and update policy θ with gradient descent   ▷ Update
    end for
end for
```

$r(p, b)$ represents the scalar output generated by ComplianceReward (parametrized by θ), given the planned workflow action p and the generated block b. The term b_w denotes the favored choice among the pair of responses, b_w and b_l, in the comparison dataset D.

The generated block b can include multiple utterances by both the user and the system. We found that there are advantages in excluding the dialogue context and presenting only b to the model due to the following reasons: (1) The model can focus more effectively on evaluating the text itself rather than being distracted by the typically longer context, (2) Constructing negative instances (b_l) becomes straightforward by replacing the workflow of the positive instance b_w with an alternative.

Data Collection. We first segment each conversation into multiple blocks s, comprising contiguous utterances that are annotated with the same workflow step p. By pairing utterances from different segments with different workflow annotation p_l such that $p \neq p_l$, we generate (p, b_w, b_l) triplets.

4.3 Compliance Optimization

Finally, the system model is updated according to a combination of the standard LM loss and a KL divergence loss between the updated model and the reference model, shown in Equation 2.

$$\begin{aligned}&\max_{\theta} \mathbb{E}_{k\sim\mathcal{U}(1,K)}\mathbb{E}_{(c,b)\sim\mathcal{D}^k}[\log l_\theta(b|c,r_k)\\ &-\beta\sum_{t=1}^{T}\mathrm{KL}\left(l_0(\cdot|b_{<t},c)\|l_\theta(\cdot|b_{<t},c,r_k)\right)]\end{aligned} \tag{2}$$

5 Experimental Setup

Datasets. We evaluate our approach using two TOD datasets, where users aim to accomplish specific tasks through dialogue. Each dataset consists of conversations between two speakers, a system or agent, and a user or customer.

- **Action Based Conversations Dataset (ABCD) (Chen et al., 2021a)**: Contains $\sim$10,000 dialogues between customers and agents and spans 55 intents. The agents have explicit workflows they need to follow according to company guidelines, making it an ideal dataset to evaluate compliance requirements.
- **MultiWOZ 2.2 (Zang et al., 2020)**: Contains over $\sim$10,000 dialogues spanning multiple domains. We designate annotated user intents as workflow actions to be predicted and include agent dialogue acts (DAs) in the context.

Evaluation. We evaluate the different approaches and baselines on a variety of metrics:

- **LLM compliance:** We automatically evaluated compliance using an LLM (prompt in Appendix A). We used a categorical labeling scheme involving two levels: 0 = 'not compliant,' and 1 = 'fully compliant.'
- **Human compliance**: For human evaluation, we randomly selected 100 generated outputs from each model (guideline in Appendix A). We used binary labeling (0, 1) for *compliance* and had three annotators rate each example. The annotators had access to the complete policy document containing guidelines for all workflow actions.
- **Human coherence**: Annotators were asked to also rate each of the same 100 examples on *coherence*, represented as a binary (0,1) label.
- **Semantic Similarity**: We measure the similarity between generated responses and the corresponding human-annotated ground truth using commonly-used similarity measures such as BLEU, Meteor, BLEURT, and BERTScore (Papineni et al., 2002; Banerjee and Lavie, 2005; Sellam et al., 2020; Zhang et al., 2020). We report the "Block" version for each, computed by taking the max between each prediction and target utterance pair over all targets and taking the average over predictions.
- **Diversity**: We measure generated response diversity using the dist-3 metric (Li et al., 2016a).
- **Workflow Accuracy**: For the ActionPlan and ComplianceOpt models that predict the next workflow action, we report the exact match accuracy of the predicted action against ground truth.

Methods & Baselines.

- **NoAction**: A simple model that only sees user and system utterances without access to completed actions or next workflow steps as done in Sodhi et al. (2023).
- **ActionAware**: Action executions are interleaved in the input alongside utterances, allowing the model to understand the history of completed workflow actions in the dialogue context. The model may implicitly learn the relationship between workflow policies and agent utterances, enabling the generation of more contextually relevant responses. This approach applies supervised learning on dialogue context augmented with prior and future actions and is similar to the SimpleTOD model (Hosseini-Asl et al., 2020).
- **ActionPlan**: The ActionPlan model goes beyond ActionAware by explicitly modeling future compliance to workflow policy guidelines. It introduces the concept of a "planned" workflow action, representing the next action that must be completed based on the policy. This planned action is incorporated into the dialogue context, and the model generates responses that align with the intended workflow. This approach treats the planned future workflow action as a latent variable in the generation process, resulting in better workflow compliance in responses.
- **Guideline**: Instead of relying on completed actions or predicted future workflow actions, the Guideline approach conditions on a fixed "standard" sequence of actions, referred to as the guideline in (Chen et al., 2021a).
- **LLM-prompting**: We use prompting and in-context learning with large language models (LLMs) to explore the option of using nat-

ural language policy guidelines as a source of workflow information. Similar to prior work (Zhang et al., 2023), our LLM prompt consists of instructions that describe the task and task-related text that consists of guidelines, example conversations, and the dialogue context C_t. Our LLM-prompting method assumes ORACLE next workflow and generates corresponding responses. We include our prompt in Appendix A.
- **PREDICTED/ORACLE** variants: At test time, our ACTIONPLAN and COMPLIANCEOPT models predict the next workflow action and condition system response generation on this action (referred to as PREDICTED). ORACLE models are supplied with ground truth next workflow actions, to gauge the upper bound of performance. Finally, **ACTIONPLANALL FUTURE ORACLE** uses all future remaining workflow steps annotated in the ground truth data.

Training. Our dialogue system models and user simulators are both initialized with pretrained DistilGPT2 (Sanh et al., 2019), which is a condensed variant of GPT-2 (Radford et al., 2019). Tokenization of inputs to the system and user models use pre-trained BPE codes (Sennrich et al., 2016). For the ComplianceReward model, we start with a pre-trained RoBERTa model, with its associated BPE tokenizer. (Liu et al., 2019). Training procedure and hyperparameters are included in Appendix A.

6 Results and Analyses

6.1 Overall Results

- *Workflow-awareness consistently improves performance:* Models incorporating workflow information show higher compliance over the NOACTION baseline (Table 5, Figure 3).
- *Direct compliance optimization leads to peak system compliance:* Our investigation reveals that COMPLIANCEOPT, which utilizes reinforcement learning to optimize compliance scores, outperforms models trained with teacher-forcing. This approach not only successfully optimizes response for the ComplianceReward model (Table 5), but also leads to high performance in human compliance and LLM-based evaluations (Figure 3).
- *Human evaluation validates automated metrics:* Human evaluators corroborate the results obtained from automated evaluations, confirming that workflow-aware models consistently outperform baselines. Remarkably, RL optimization achieves higher compliance without compromising coherence (Figure 3).
- *Consistent performance across datasets:* The improved performance of workflow-aware models, particularly ACTIONPLAN and COMPLIANCEOPT, extends beyond the primary ABCD dataset. These findings hold even when validated on more general task-oriented dialogue datasets, such as MultiWoz (Table 3).
- *Ablation studies:* We also conduct ablation studies to investigate the effectiveness of explicitly predicting workflow actions compared to directly following standardized workflow guidelines. We also explore the impact of predicting and conditioning on future action sequences as opposed to single actions (Tables 5,2).
- *Workflow-aware models maintain high performance on dialogue metrics:* While our approach generates responses directly without predicting intermediate slot values, we include Action State Tracking (AST) dialogue metrics in Appendix A.2 by extracting these values from the response. We show that we can perform comparably to prior work on these metrics.

6.2 Compliance

First, we evaluate all approaches on how well they generate compliant responses. We conduct human and LLM evaluations. We also report ComplianceReward on the test dataset and show that it correlates with both LLM and human scores.

LLM evaluation: We first evaluate compliance using the LLM evaluation prompt. The results, shown on the leftmost plot in Figure 3, indicate that ground truth responses achieve the highest compliance and workflow-aware models outperform baselines. COMPLIANCEOPT achieves higher compliance scores compared with ACTIONPLAN, which highlights the benefit of directly optimizing for compliance. Also, the ORACLE model variants perform better than their PREDICTED counterparts, which means that we can generate more compliant responses when we have access the true next workflow action.

	ACTIONAWARE	ACTIONPLAN PREDICTED	COMPLIANCEOPT PREDICTED	Ground Truth
Avg Compliance	0.31	0.29	**0.35**	0.46
At least 1 Compliant (↑)	0.41	**0.53**	0.47	0.57
All Compliant (↑)	0.2	0.19	**0.24**	0.38
Fleiss Kappa	0.66	0.65	0.67	0.73

Table 1: Human Annotation Metrics & Inter-annotator Agreement (full table in Appendix)

Subflow / Action	status_credit_missing / ask-the-oracle	
Summarized Context	A client is asking the agent for help with 40 dollars of missing credit.	
Policy Guideline	Verify the client's claim by clicking 'Asking the Oracle'.	
Model	Response	Reward
ActionAware	**Agent**: i will reissue a credit for the $40	0.4857
ActionPlan Predicted	**Agent**: i just checked the system and they are showing up **Client**: oh, okay. thanks! **Agent**: how much were the credit?	0.7608
ComplianceOpt Predicted	**Agent**: let me check the system to see if there is a problem with your account	0.9690
ActionPlan Oracle	**Agent**: what is the issue? **Client**: i never bought anything. **Agent**: ok thanks	0.7536
ComplianceOpt Oracle	**Agent**: let me check the system	0.9743
Ground Truth	**Agent**: i am going to check if this is an error on our side now. **Client**: ok thank you	0.9547

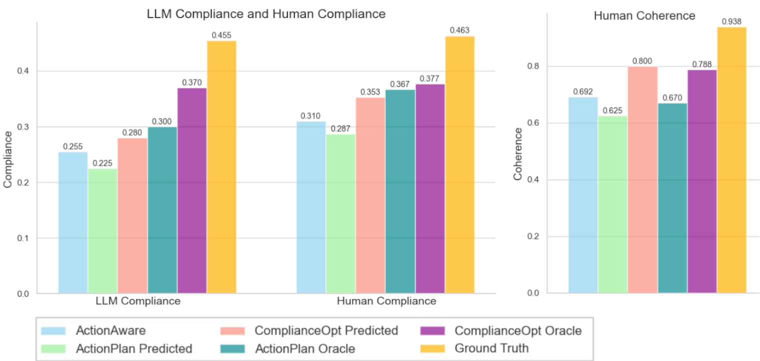

Figure 3: **Left**: Sample Simulated Interaction between Agent Models and User Simulator. **Right**: Evaluation results with human annotators and LLM. We report the average score received for each model.

Human evaluation: Next, we evaluate compliance using human evaluation. In the middle plot in Figure 3, we show the average compliance score across all annotators for each approach. Results show that ComplianceOpt models are more compliant than their ActionPlan counterparts, in both Predicted and Oracle variants. Additionally, we observe similar trends for both human and LLM compliance judgments, with ground truth responses receiving the highest scores, trailed by the Oracle models, then by the Predicted models.

Table 1 shows further breakdown of the human-annotated scores. We compute the percentage of examples that received at least 1 compliant score and the percentage that received all compliant scores. In this analysis, we observe a similar trend, with ground truth performing best, followed by ComplianceOpt, which outperforms ActionPlan on "all compliant" and "at least 1 non-compliant". We used Fleiss' Kappa for assessing annotator agreement. We find that the annotators are in "substantial agreement", showing that human compliance judgment is a reliable metric for compliance evaluation (Landis and Koch, 1977).

ComplianceReward While LLM and human evaluation are the most reliable way to evaluate compliance, we also report ComplianceReward, our training reward signal, on the test dataset. We compute the reward values across different approaches (Table 5). We find the compliance reward to be positively correlated with both human (Table 6) and LLM evaluation (Table 7). More details are included in Appendix A.2.

6.3 Coherence

In addition to compliance, we also evaluated whether generated responses were coherent. As shown in the rightmost plot in Figure 3, ground truth responses have the highest coherence scores, followed closely by both variants of ComplianceOpt. This indicates that ComplianceOpt is able to achieve higher compliance scores while also maintaining high fluency and coherence.

6.4 Semantic Similarity and Diversity

Model	Block BertScore	Block BLEURT	Block METEOR	Block BLEU	dist-3	Workflow Accuracy
Baselines & Ablations						
NoAction	0.8577	0.2286	0.0549	0.4481	0.7738	N/A
Guideline Oracle	0.8679	0.2763	0.0679	0.4928	0.7536	N/A
LLM-prompting Oracle	0.8676	0.3933	0.0609	0.5493	0.7013	N/A
ActionPlan All Future Oracle	0.8560	0.2498	0.0535	0.4606	0.7479	N/A
Proposed Methods						
ActionAware	0.8642	0.2703	0.0726	0.4745	0.7661	N/A
ActionPlan Predicted	0.8685	0.2959	0.0808	0.4951	**0.7707**	**0.7011**
ComplianceOpt Predicted	**0.8740**	**0.2964**	**0.0924**	**0.5075**	0.6553	0.6821
ActionPlan Oracle	0.8683	0.3081	0.0881	0.5021	**0.7683**	N/A
ComplianceOpt Oracle	**0.8745**	**0.3312**	**0.1156**	**0.5287**	0.6591	N/A
Ground Truth	N/A	N/A	N/A	N/A	0.7738	N/A

Table 2: Semantic Similarity and Diversity Results

ActionPlan and ComplianceOpt achieve the highest semantic similarity scores when compared with ground truth-compliant responses (Table 2). This result indicates that adding future planned actions can lead to more contextually relevant and compliant system responses. Moreover, ActionPlan and ComplianceOpt Oracle models outperform their Predicted counterparts, which suggests that using the *true* next workflow action results in responses more aligned with the human-annotated compliant behavior.

In addition, the lower dist-3 scores obtained by the ComplianceOpt models, regardless of whether they are in the Predicted or Oracle configuration, suggest that these models produce less diverse responses. One explanation is that the COMPLIANCEOPT models, as a result of the RL-based optimization, learn to focus on generating a narrower range of utterances that are compliant given the context. Since the ground truth responses achieve both higher dist-3 and compliance rewards, this effect seems unique to the RL optimization.

6.5 Ablations

Effect of including all future actions. Since including future workflow actions results in more compliant responses, we explore if adding *all* future workflow actions would result in even more compliant behavior (ACTIONPLAN ALL FUTURE ORACLE vs. ACTIONPLAN ORACLE). Table 5 shows that including all future actions can hurt performance, likely because too much future information leads to noise and model confusion. In contrast, simply focusing on the next workflow action leads to compliant localized interactions ("blocks").

Training a model with standardized workflows. We consider the effect of conditioning on standardized workflows, without dynamically including the next workflow actions in the context. As shown in Table 5, the GUIDELINE ORACLE model performs better than the baseline but worse than workflow-aware models because it does not dynamically generate contextually relevant workflow actions and responses. This reinforces the importance of dynamic workflow prediction, which captures the inherent uncertainty in dialogues.

Few-shot LLM prompting with workflow guidelines. The final model variant we considered was directly using an LLM to predict the next workflow action, instead of fine-tuning a separate model. The LLM-PROMPTING ORACLE model achieves the second-highest compliance reward after the COMPLIANCEOPT ORACLE. We see that the COMPLIANCEOPT model, explicitly trained to optimize compliance can outperform or match the LLM with orders of magnitude more parameters (`gpt-3.5-turbo`). The high text similarity scores achieved by the LLM-PROMPTING ORACLE, often outperforming even the best-performing ACTIONPLAN and COMPLIANCEOPT models in terms of metrics like BLEURT and BLEU, validate the value of using guided prompts to improve response compliance. We note that RL optimization of LLMs requires much larger computational resources and remains an interesting future work.

Model	Compliance Score	BertScore	BLEURT	METEOR	BLEU	dist-3
Baseline						
NOACTION	0.7446	0.4711	0.2476	0.0959	0.0108	0.4366
Proposed Methods						
ACTIONAWARE	0.8451	0.8575	0.4001	0.1959	0.0252	0.8086
ACTIONPLAN PREDICTED	0.8463	0.8496	0.3928	**0.1936**	0.0249	0.8027
COMPLIANCEOPT PREDICTED	**0.8853**	**0.8616**	**0.4310**	0.1917	**0.0265**	**0.8267**
ACTIONPLAN ORACLE	0.8573	0.8506	0.3897	0.1900	0.0242	0.7962
COMPLIANCEOPT ORACLE	**0.9153**	**0.8622**	**0.4271**	**0.1951**	**0.0278**	**0.8137**
Ground Truth	0.8946	N/A	N/A	N/A	N/A	0.8237

Table 3: Automated Evaluation Results on MultiWOZ 2.2. PREDICTED variants of ACTIONPLAN and COMPLIANCEOPT achieved 69% and 75% workflow accuracy respectively.

6.6 MultiWOZ Experiment Results

In our MultiWOZ experiments, we find consistent support for our approach. Workflow-aware models, particularly ACTIONPLAN and COMPLIANCEOPT, outperform NOACTION and ACTIONAWARE in both PREDICTED and ORACLE settings, showcasing their capacity to generate compliant and contextually relevant responses.

However, there are several differences when compared to the ABCD experiments. MultiWOZ introduces increased response diversity, especially noticeable in the COMPLIANCEOPT models, a departure from the ABCD dataset's behavior. Moreover, workflow-aware models benefit significantly from action annotation, as seen in the NOACTION versus ACTIONAWARE comparison. We conjecture that these disparities may be attributed to differences in action annotation and the nature of actions, which are typically resolved in a single interaction in MultiWOZ, in contrast to the more intricate workflows in the ABCD dataset.

7 Conclusion

In this paper, we propose the problem of workflow-guided response generation and introduce a novel RL-based framework to train workflow-compliant models for task-oriented dialogue. By integrating workflow information during training and directly optimizing for compliance, our approach improves upon baseline models and generates responses that are both workflow-compliant and linguistically natural. We evaluate our models on both ABCD and MultiWoz datasets and show empirical improvements in automated and human evaluation metrics.

8 Limitation

This paper introduced a novel RL-based framework that generates workflow-guided responses for task-oriented dialogue. While this is promising, several limitations warrant discussion. **Higher fidelity user simulator models:** Fully simulating real-world customer service interactions would provide a more comprehensive evaluation. For instance, our user model was intentionally kept simple to facilitate the development and testing. Using more sophisticated models that incorporate diverse user behaviors can potentially help with better generalization. Using the same models for users and agents can also increase efficiency. **Workflow requirements:** We consider dialogue settings where workflow information is available, e.g. policy guidelines in customer-service interactions, and indeed more useful to follow compared to slot-value objectives. However, for datasets that do not have explicit workflows, one would have to proxy workflows, e.g. in MultiWoz we had to create workflows from from user intent and system act annotations. This can limit the applicability of our method.

References

Vevake Balaraman, Seyedmostafa Sheikhalishahi, and Bernardo Magnini. 2021. Recent neural methods on dialogue state tracking for task-oriented dialogue systems: A survey. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 239–251, Singapore and Online. Association for Computational Linguistics.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Iñigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. MultiWOZ - a large-scale multi-domain Wizard-of-Oz dataset for task-oriented dialogue modelling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5016–5026, Brussels, Belgium. Association for Computational Linguistics.

Bill Byrne, Karthik Krishnamoorthi, Chinnadhurai Sankar, Arvind Neelakantan, Ben Goodrich, Daniel Duckworth, Semih Yavuz, Amit Dubey, Kyu-Young Kim, and Andy Cedilnik. 2019. Taskmaster-1: Toward a realistic and diverse dialog dataset. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4516–4525, Hong Kong, China. Association for Computational Linguistics.

Derek Chen, Howard Chen, Yi Yang, Alex Lin, and Zhou Yu. 2021a. Action-based conversations dataset: A corpus for building more in-depth task-oriented dialogue systems. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021*, pages 3002–3017, Online. Association for Computational Linguistics.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Michael Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. 2021b. Decision transformer: Reinforcement learning via sequence modeling. *CoRR*, abs/2106.01345.

Mehdi Fatemi, Layla El Asri, Hannes Schulz, Jing He, and Kaheer Suleman. 2016. Policy networks with two-stage training for dialogue systems. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 101–110, Los Angeles. Association for Computational Linguistics.

Milica Gašić, Catherine Breslin, Matthew Henderson, Dongho Kim, Martin Szummer, Blaise Thomson, Pirros Tsiakoulis, and Steve Young. 2013. POMDP-based dialogue manager adaptation to extended domains. In *Proceedings of the SIGDIAL 2013 Conference*, pages 214–222, Metz, France. Association for Computational Linguistics.

Amine El Hattami, Stefania Raimondo, Issam Hadj Laradji, David Vázquez, Pau Rodríguez López, and Christopher Joseph Pal. 2022. Workflow discovery from dialogues in the low data regime. *Trans. Mach. Learn. Res.*, 2023.

Wanwei He, Yinpei Dai, Yinhe Zheng, Yuchuan Wu, Zheng Cao, Dermot Liu, Peng Jiang, Min Yang, Fei Huang, Luo Si, et al. 2022. Galaxy: A generative pre-trained model for task-oriented dialog with semi-supervised learning and explicit policy injection. *Proceedings of the AAAI Conference on Artificial Intelligence*.

Matthew Henderson, Blaise Thomson, and Jason D. Williams. 2014. The second dialog state tracking challenge. In *Proceedings of the 15th Annual Meeting of the Special Interest Group on Discourse and Dialogue (SIGDIAL)*, pages 263–272, Philadelphia, PA, U.S.A. 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. In *Advances in Neural Information Processing Systems*, volume 33, pages 20179–20191. Curran Associates, Inc.

Youngsoo Jang, Jongmin Lee, and Kee-Eung Kim. 2022. Gpt-critic: Offline reinforcement learning for end-to-end task-oriented dialogue systems. In *International Conference on Learning Representations (ICLR)*.

J. Richard Landis and Gary G. Koch. 1977. The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.

Chia-Hsuan Lee, Hao Cheng, and Mari Ostendorf. 2021. Dialogue state tracking with a language model using schema-driven prompting. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4937–4949, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mike Lewis, Denis Yarats, Yann Dauphin, Devi Parikh, and Dhruv Batra. 2017. Deal or no deal? end-to-end learning of negotiation dialogues. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2443–2453, Copenhagen, Denmark. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California. Association for Computational Linguistics.

Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016b. Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Austin, Texas. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Ximing Lu, Sean Welleck, Jack Hessel, Liwei Jiang, Lianhui Qin, Peter West, Prithviraj Ammanabrolu, and Yejin Choi. 2022. QUARK: controllable text generation with reinforced unlearning. In *NeurIPS*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *NeurIPS*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Olivier Pietquin, Matthieu Geist, Senthilkumar Chandramohan, and Hervé Frezza-Buet. 2011. Sample-efficient batch reinforcement learning for dialogue management optimization. *ACM Trans. Speech Lang. Process.*, 7(3).

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Stefania Raimondo, Christopher Pal, Xiaotian Liu, David Vazquez, and Hector Palacios. 2023. Improving generalization in task-oriented dialogues with workflows and action plans.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8689–8696. AAAI Press.

Verena Rieser and Oliver Lemon. 2011. *Reinforcement Learning for Adaptive Dialogue Systems: a data-driven methodology for Dialogue Management and Natural Language Generation*.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *NeurIPS EMC2 Workshop*.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Igor Shalyminov, Alessandro Sordoni, Adam Atkinson, and Hannes Schulz. 2020. Fast domain adaptation for goal-oriented dialogue using a hybrid generative-retrieval transformer. In *2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.

Satinder Singh, Diane Litman, Michael Kearns, and Marilyn Walker. 2002. Optimizing dialogue management with reinforcement learning: Experiments with the njfun system. *J. Artif. Int. Res.*, 16(1):105–133.

Charlie Snell, Ilya Kostrikov, Yi Su, Sherry Yang, and Sergey Levine. 2023. Offline RL for natural language generation with implicit language Q learning. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Paloma Sodhi, Felix Wu, Ethan R. Elenberg, Kilian Q Weinberger, and Ryan Mcdonald. 2023. On the effectiveness of offline RL for dialogue response generation. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 32088–32104. PMLR.

Yixuan Su, Lei Shu, Elman Mansimov, Arshit Gupta, Deng Cai, Yi-An Lai, and Yi Zhang. 2022. Multi-task pre-training for plug-and-play task-oriented dialogue system. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4661–4676, Dublin, Ireland. Association for Computational Linguistics.

Ryuichi Takanobu, Hanlin Zhu, and Minlie Huang. 2019. Guided dialog policy learning: Reward estimation for multi-domain task-oriented dialog. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 100–110, Hong Kong, China. Association for Computational Linguistics.

Wei Wei, Quoc Le, Andrew Dai, and Jia Li. 2018. AirDialogue: An environment for goal-oriented dialogue research. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3844–3854, Brussels, Belgium. Association for Computational Linguistics.

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. In *Proceedings of the 2nd Workshop on Natural Language Processing for Conversational AI*, pages 109–117, Online. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Xiaoying Zhang, Baolin Peng, Jianfeng Gao, and Helen Meng. 2022. Toward self-learning end-to-end task-oriented dialog systems. *arXiv preprint arXiv:2201.06849*.

Xiaoying Zhang, Baolin Peng, Kun Li, Jingyan Zhou, and Helen M. Meng. 2023. Sgp-tod: Building task bots effortlessly via schema-guided llm prompting. *ArXiv*, abs/2305.09067v1.

A Appendix

A.1 Human Annotation Metrics & Inter-annotator Agreement

We include the complete human annotation metrics and inter-annotator agreement in Table 4.

	ActionAware	ActionPlan Predicted	ComplianceOpt Predicted	Ground Truth
Avg Compliance	0.31	0.29	**0.35**	0.46
At least 1 Compliant (↑)	0.41	**0.53**	0.47	0.57
All Compliant (↑)	0.2	0.19	**0.24**	0.38
At least 1 Non-Compliant (↓)	0.79	0.8	**0.75**	0.62
All Non-Compliant (↓)	0.58	**0.47**	0.53	0.42
Fleiss Kappa	0.66	0.65	0.67	0.73

Table 4: Complete Human Annotation Metrics & Inter-annotator Agreement

A.2 ComplianceReward Analysis

Model	Compliance Reward
Baselines & Ablations	
NoAction	0.4963
Guideline Oracle	0.5713
LLM-prompting Predicted	0.6421
LLM-prompting Oracle	0.8410
ActionPlan All Future Oracle	0.6043
Proposed Methods	
ActionAware	0.6012
ActionPlan Predicted	**0.6762**
ComplianceOpt Predicted	0.6742
ActionPlan Oracle	0.7925
ComplianceOpt Oracle	**0.8670**
Ground Truth	0.8676

Table 5: ComplianceReward Results

Table 5 shows that our proposed framework, as expected, outperforms other methods on this metric. We also computed the correlation between ComplianceReward scores and human evaluation (Table 6) on the human-annotated subset of our model-generated responses and found that there is a positive correlation. There is a similar positive correlation between ComplianceReward and LLM compliance (Table 7). This validates that ComplianceReward as a training signal is indeed well correlated with human measures of compliance. Moreover, the model performances across human and LLM scores are generally similar to that of the ComplianceReward rankings, with GroundTruth and ComplianceOpt receiving the highest scores.

	Annotator Avg Score	Model Avg Score	Pearson	Spearman
Ground Truth	0.4631	0.6290	0.2414	0.2437
ACTIONAWARE	0.3100	0.4349	0.4075	0.4146
ACTIONPLAN PREDICTED	0.2867	0.5416	0.2114	0.1645
COMPLIANCEOPT PREDICTED	0.3533	0.6394	0.2819	0.2702

Table 6: Comparison of the ComplianceReward with human judgment. All correlations are significant with $p < 0.05$.

	LLM Score	Model Avg Score	Pearson	Spearman
Ground Truth	0.4552	0.6290	0.3526	0.5030
ACTIONAWARE	0.2553	0.4349	0.3012	0.3549
ACTIONPLAN PREDICTED	0.2254	0.5416	0.4362	0.5502
COMPLIANCEOPT PREDICTED	0.2802	0.6394	0.3252	0.4256

Table 7: Comparison of the ComplianceReward with LLM compliance. All correlations are significant with $p < 0.05$.

A.2.1 Action State Tracking Evaluation

While our framework focuses on response generation and does not directly predict slot values, we extract these values from the generated responses and compute model performances on Action State Tracking (AST) metrics, which is a set of performance benchmark metrics proposed for the ABCD dataset (Chen et al., 2021a). Specifically, we adopt the approach of Lee et al. (2021) that has competitive performance in extracting slot values from generated responses and train a `t5-base` model to extract slot values from generated system responses.

We additionally report the workflow accuracy performances of the PREDICTED versions of ACTIONPLAN and COMPLIANCEOPT models for comparisons. The PREDICTED models first predict the next planned workflow action and condition the next response generation on the predicted action.

From Table 8, we find that ground truth and our models perform similarly on both b-slot and value predictions. We note that our framework does not optimize these metrics, and this evaluation shows that our optimization does not lead to a strong degradation of the standard dialogue metrics.

A.3 User Simulator

We instantiate the user simulator with the NOACTION model.

	Workflow Accuracy	B-Slot	Value	Action (Joint)
Ground Truth	1	0.5737	0.5789	0.4895
COMPLIANCEOPT PREDICTED	0.7011	0.5632	0.6000	0.5105
COMPLIANCEOPT ORACLE	N/A	0.5526	0.6211	0.4895
ACTIONPLAN PREDICTED	0.6821	0.5526	0.5632	0.4684
ACTIONPLAN ORACLE	N/A	0.5632	0.5474	0.4579
ACTIONAWARE	N/A	0.5632	0.5789	0.4789

Table 8: Action State Tracking Metrics Results on the ABCD dataset.

A.4 Experiment Details

A.4.1 Block Processing

To generate workflow blocks described in Section 3.1 for training the ComplianceReward model, we use the following simple approach:

- For each conversation in the dataset, we traverse the utterances in reverse order, starting from the final turn.
- At each moment of the traversal, we maintain an index of the latest completed workflow action. We begin by initializing the latest completed action as "Send-off/Goodbye", which is a placeholder to account for the final moments of the dialogue where no action is completed.
- When a new completed action is seen, the index is updated. Moreover, agent utterances between the old index and the new action index are marked with the old workflow action, and the marked utterances are considered as making up a “block“ of exchanges, which then can be used for training the ComplianceReward model.

A.4.2 Input Formatting

For both training and inference, ACTIONPLAN and COMPLIANCEOPT models use a simple input format (Figure 4) which interleaves dialogue history, planned action, and completed actions, similar to that of SimpleTOD (Hosseini-Asl et al., 2020). NOACTION and ACTIONAWARE models use variants of the format, each without planned and completed action, and completed actions, respectively.

A.4.3 Training Procedure

Our method comprises two stages: teacher-forcing training and RL training with direct compliance

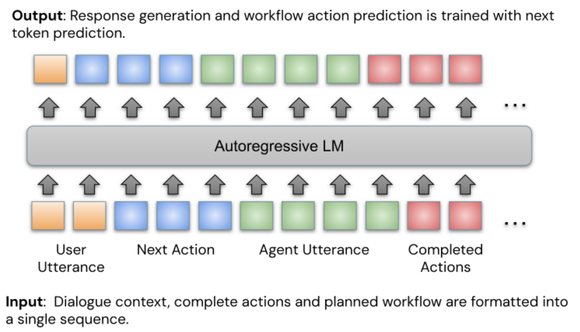

Figure 4: Our framework uses an autoregressive LM with interleaved utterances, actions, and workflow actions as the input.

optimization. ActionAware, ActionPlan use teacher-forcing training with different workflow action inputs while the third (ComplianceOpt) uses RL training with direct compliance optimization. ActionAware conditions only on completed actions $\{a_t\}$, while ActionPlan additionally conditions on predicted future actions. In teacher-forcing training (ActionAware, ActionPlan), the LM is trained with the standard negative log-likelihood loss, while in RL training, it is trained based on a reward signal generated by the ComplianceReward model.

ActionAware. A simple way to include workflow action information is to include the history of past workflow actions in the dialogue context C_t. In this way, an action execution a_t is treated as an utterance, and indicates that the system completed a workflow action at time t. Thus, an example context might be $C_t = [u_0, s_1, u_2, s_3, a_4, \cdots]$. Conditioned on C_t, the LM can then generate the system utterance s_{t+1}. This implicitly models the relationship between workflow policy and agent utterances, since the LM may learn to use patterns between completed actions and the next system response.

ActionPlan. Including only the previously completed actions does not directly model future compliance with policy guidelines. To explicitly model this, we introduce a *future workflow action*, as the next workflow action that must be completed. Given a completed ground truth dialogue, we construct this input sequence by backpropagating the action execution a_t to all previous utterances $\{s_k\}$, $k < t$, before another action occurs. We define this action assignment as a *planned* workflow action p_t associated with every system utterance s_t. We also include past actions in the context to help model workflow dynamics. An example context is $C_t = [u_0, (p_1, s_1), u_2, (p_3, s_3), a_4, \cdots]$, where $p_1 = p_3 = a_4$.

ComplianceOpt. Next, we apply reinforcement learning to directly optimize workflow compliance. Specifically, a teacher-forcing trained ActionPlan model is used as an initial model for this step and uses the same input formatting. To apply the interactive sampling model to generate a block of utterance, we use a frozen copy of a teacher-forcing trained response generator.

At each training step, the agent and user models simulate a "block" session by generating one turn at a time. A turn is defined as a consecutive sequence of utterances made by the same party. A block session is terminated either when the agent completes the planned action, or the turn count reaches a predefined limit. When a session is terminated, the dialogue history and the simulated exchange are then scored by the ComplianceReward model, which are then quantized and trained according to the modified Quark algorithm (Algorithm 1).

A.4.4 Inference

The inference procedure for generating a block of interaction between the user and agent models is identical to the interactive sampling step used for optimizing the ComplianceOpt model.

A.4.5 Parameters & Hyperparmeters

We list the parameters and hyperparameters we used for our experiments in Table 9. We tune our learning rate, interactive sampling temperature, and the number of quantiles K using grid search.

Teacher-Forcing Setting	
Agent model	`distilgpt2`
LLM prompting model	`gpt-3.5-turbo`
Scoring model detail	`roberta-base`
Training epochs	10 (ABCD) / 1 (MultiWOZ)
Learning Rate	2e-5
Special tokens	START_USER, START_WORKFLOW, END_WORKFLOW, START_AGENT, END_AGENT, START_ACTION, END_ACTION, START_DIALOG, END_DIALOGUE
Compliance Optimization RL Setting	
Agent model	`distilgpt2` (Warm-start ActionPlan)
Client model	`distilgpt2`
Learning Rate	2e-5
Sampling Temperature	0.5
Number of Interactions	3
Number of Quantiles K	5
KL weight β	0.05
Training Steps	80k (ABCD) / 160k (MultiWOZ)

Table 9: Experiment Models & Parameters

A.5 LLM Prompts

```
generation_prompt = f"You are a
    cusotmer agent helping a customer
    with a issue. Read the dialogue
    context, provided policy
    guideline, and generate an agent
    utterance to help the customer in
    a way that is compliant to the
    guideline. The generated agent
    turn should be at most 2
    utterances, and should be similar
    in length to the agent utterances
    shown in the examples that
    demonsrtate compliant agent
    behavior.\n\Custome Situation:
    {s}\n\Policy Action Name:
    {w}\n\Policy Name Guideline:
    {g}\n\n\{example_str}dialogue
    Context: {i}\n\n\Agent: "
```

```
evaluation_prompt = f"Read the provide
    guideline and assess the extent to
    which the agent's behavior in the
    input interaction aligns with the
    specified workflow action,
    considering the name and a concise
    description of the workflow
    provided. 1 = Compliant\n0 =
    Non-compliant\n\nSubflow:
    {s}\nWorkflow: {w}\nDescription:
    {g}\n\n\Dialogue
    History:\n{i}\n\nInput
    Interaction:\n{r}\n\nAnswer:"
```

A.6 Human Evaluation Guidelines (Abridged)

Compliance: Assess if the agent's behavior aligns with the specified workflow action, taking into account the action's name and policy guideline. If the agent has already completed certain steps or the entire policy guideline behavior in the dialogue history, they should not be penalized for not repeating those corresponding steps.

Coherence: Rate the coherence of the agent's interaction on a binary scale (0=not coherent, 1=coherent). In this evaluation, please do not consider repetitive agents as coherent. Additionally, do not include incoherent or disfluent client behavior in the evaluation (only evaluate agent behavior).

A.7 Human Evaluation Guidelines (Full)

```
Agent Quality Annotation Task
Task
Evaluate the agent's performance in an
    interaction with a customer based
    on a set of categories to gain
    insights into various aspects of
    agent behavior. The primary
    category to consider is workflow
    compliance, which determines if
    the agent has successfully
    achieved the objectives outlined
    in the provided workflow action
    during their interaction with the
    client. The other category is
    coherence.

Relevant Documents
ABCD Guideline: This document consists
    of comprehensive descriptions for
    each customer assistance subflow,
    including specific action steps
    within each subflow.

To locate a particular workflow
    action, begin by referring to the
    relevant subflow section (e.g.,
    Initiate Refund), and then
    identify the corresponding
    workflow action enclosed within
    brackets (e.g., [Pull Up Account]).

In the annotation sheet, we will also
    provide brief policy guidelines
    alongside examples to aid in the
    annotation process. Ideally,
    policy guidelines should be
    sufficient for the annotations.

Categories
1. Compliance
Assess the degree to which the agent's
    behavior aligns with the specified
    workflow action, taking into
    account the action's name and
    policy guideline. Please refer to
    the provided document for more
    detailed information. If the agent
    has already completed certain
    steps or the entire policy
    guideline behavior in the dialogue
    history, they should not be
    penalized for not repeating those
    corresponding steps.
```

1 = Compliant: The agent successfully executes all the steps outlined in the policy guideline.
0 = Non-compliant: The agent fails to execute any of the steps mentioned in the policy guideline.

Examples:
0 = Non-compliant
Subflow: out_of_stock_general
Workflow Action: notify-team
Policy Guideline: Let the customer know that you will write up a report and let the Purchasing Department know about this, so they can do a better job.

Enter 'purchasing department' into the input box and [Notify Internal Team]
Dialog History: Omitted Here
Target Generation
Agent: i m sorry for your inconvenience
Client: ok
Agent: i can offer you a promo code

1=Compliant:
Subflow: out_of_stock_general
Workflow Action: notify-team
Policy History: Let the customer know that you will write up a report and let the Purchasing Department know about this, so they can do a better job.

Enter 'purchasing department' into the input box and [Notify Internal Team]
Dialog Context: Omitted Here
Target Generation
Agent: okay, i'm sorry for this situation, what item is out of stock?
Client: product: jacket brand: michael_kors amount: $49
Agent: the most i can do for you is write up a report to the purchasing department about this. would that be alright?

Subflow: shipping status
Workflow Action: update-order
Description: If the Oracle says No, then the customer will not be happy. To resolve, enter how you will fix the problem

Options include: 'change date', 'change address', 'change item', or 'change price'
Enter into [Update Order]
Dialog Context: Omitted Here
Target Generation
Agent: it seems the email was incorrect. when were you expecting it to arrive?
Client: tomorrow pm.
Client: i'm just trying to verify if it is still coming or something weird happened.
Agent: okay, i can change the date to be sure you receive by tomorrow.
Client: ok, that's fine, thanks.

2. Coherence
Please rate the coherence of the agent's interaction on a binary scale (0=not coherent, 1=coherent). In this evaluation, please do not consider repetitive agents as coherent. Additionally, do not include incoherent or disfluent client behavior in the evaluation (only evaluate agent behavior).

Coherence Standard: Are all of the system responses in the conversation coherent with respect to each other?

Examples:
0=not coherent (agent repeats oneself, contradicts oneself, or says illogical things):
Agent: thank you albert. may i also have your order id?
Client: it was 9940404040

Agent: thank you albert. may i also have your order id?
Client: sure, it's 9940404040
Agent: thank you albert. may i also have your account id and order id?

Agent: how much was the service?
Client: it was $40.
Agent: how much was the extra price?
Client: i was charged $40
Agent: how much was the price you were charged?

Agent: unfortunately, since the order has already been shipped i can not cancel the order. since it has already been shipped i can only cancel the order.
Client: ok, thank you
Agent: i apologize for the inconvenience
Agent: since the order has already been sent i can not cancel the shipment.
Client: ok, i understand
Client: that is all i needed
Agent: great. have a nice day!

1=coherent:
Agent: ok, i see your refund is in progress and it looks like it should be going through to completion later today or by tomorrow at the latest
Client: okay, thank you
Agent: you're welcome
Agent: can i help with anything else?
Client: that will be all
Agent: ok, have a good day

Agent: thanks for your information.
Agent: the system said that your shipping address is the same as the one you stated above. the email was incorrect. you can ignore it.
Client: thank you that's all i needed to know
Agent: great, is there anything else that i can help you with?
Client: no, that is all.
Agent: have a nice day!