Enhancing Large Language Model Powered Task-Oriented Dialogue Systems Through Look-Forward Motivated Goals

Anonymous ACL submission

Abstract

The task-oriented dialogue (ToD) systems aim to achieve the dialogue goal **efficiently** and **successfully** in multiple turns. Additionally, the development of large language models (LLMs) has significantly enhanced the question answering and dialogue generation, and makes them become increasingly popular in current practical scenarios. Unfortunately, existing LLM-powered ToD systems lack direct rewards toward the final dialogue goal and do not account for proactivity in dialogue, which can enhance efficiency. To fill this gap, we introduce the **ProToD** (Proactively Goal-Driven LLM-powered ToD) approach, which anticipates future dialogue actions and incorporates goal-oriented reward signals to enhance ToD systems. Additionally, we present a novel evaluation method that assesses ToD systems based on goal-driven dialogue simulations. This method allows us to gauge user satisfaction, system efficiency and success rate while overcoming the limitations of current Information and Success metrics. We conduct empirical experiments on the MultiWoZ 2.1 and SGD dataset. Especially, results on the MultiWoZ 2.1 dataset demonstrate that our model achieves superior performance using only 10% of the data compared to previous end-to-end fully supervised models. This improvement is accompanied by enhanced user satisfaction and dialogue efficiency.

1 Introduction

A task-oriented dialogue system is designed to assist users in achieving specific objectives. Its primary focus is on comprehending user needs and generating responses that help to satisfy their needs. Integral to this process is the system's proactivity — an attribute borrowed from organizational behavior as described by Grant and Ashford (2008). The proactivity of conversational agents can be characterized as their ability to steer or control a dialogue toward the final goal. This is achieved by taking the initiative and foreseeing potential impacts on themselves or users. The ultimate success of a ToD system lies in taking proactive actions to effectively and efficiently address user needs.

Current research focuses on guiding LLMs to produce relevant responses using task-specific instructions and few examples. Li et al. (2023) introduce a method where a small model provides directional prompts for each query. The LLM then uses these prompts and previous dialogues as input to produce their responses. They optimize LLMs for ToD by adjusting a policy model, which can be refined using supervised learning and reinforcement based on BLEU score rewards. Additionally, Hu et al. (2023) presented a framework that uses LLMs as user simulators to enhance task-oriented dialogue models. These methods based on reinforcement approach primarily employ rewards that hinge on BLEU scores or user simulations, which focuses on assessing the similarity between the system's generated responses and predefined ground truth, as well as measuring user satisfaction scores.

However, existing metrics are limited when guiding LLMs to be proactive. Notably, the success rate is a pivotal metric in evaluating the effectiveness of a ToD system. A higher success rate indicates that the system is adept at meeting user requirements. Additionally, efficiency is gauged by the number of turns in a conversation. Fewer turns signify greater efficiency, underscoring the need for the system to be proactive. The aforementioned works do not incorporate considerations of task success and efficiency into their reward design. Moreover, LLMs tend to produce more flexible and longer responses compared to end-to-end models, leading to lower BLEU score compared with ground-truth response. When BLEU scores are used as rewards for model tuning, it can easily result in lower reward values and potentially lead to incorrect optimization directions. Therefore, a new reward mechanism that focuses on goal-driven behavior to guides LLM for

Figure 1: An example of dialogue response generation when considering future dialogues and dialogue goal

generating response is needed. If the chatbot can anticipate what the user is likely to discuss next after receiving its response, it can provide a response that smoothly connects the past and future elements of the conversation.

To tackle the outlined challenges, where task success and efficiency crucial to proactive dialogue are neglected in guiding LLM-powered ToD systems, leading to less effective optimization, we introduce the **ProToD** (Proactively Goal-Driven LLM-powered ToD) approach. This method includes two key components: **Anticipating Future Actions** and **Goal-oriented Reward**, aiming to enhance the system's effectiveness and proactivity. As depicted in Figure 1, the ability of a ToD system to anticipate users' future demands and prepare a more diverse and comprehensive response can be highly advantageous. This approach goes beyond merely predicting the next user utterance action, making it easier for the ToD system to meet users' requirements, elevate the success rate of achieving goals, and enhance overall dialogue efficiency. Furthermore, in contrast to using metrics like BLEU scores or user simulation scores as the basis for reward calculation, opting for the degree of goal completion as the reward offers a more natural and rational approach to optimizing LLM-powered ToD systems. In this context, we introduce a novel reward calculation method that considers the extent to which each system response fulfills sub-goals, replacing the previous reliance on semantic similarity or user feedback-based reward functions.

Moreover, the current metrics, such as "Inform" and "Success", which rely on fixed ground-truth values, lack the flexibility to accurately gauge effectiveness and success rates. In addition, prior research (Wu et al., 2023a) has highlighted that very high values of these metrics can be achieved simply giving a fixed and predefined response in every turn. This suggests the insufficiency of these metrics. As a result, we propose a novel evaluation method that employs GPT-4 (OpenAI, 2023) as the user simulator. In this approach, users are required to adhere to predefined goals when interacting with the ToD system. The extent to which these conversations successfully achieve their goals and the number of turns required are used to measure both success and efficiency.

To summarize, our contributions in this work are:

- We propose the **ProToD** (Proactively Goal-Driven LLM-powered ToD) approach which anticipates future dialogue actions and integrates a goal-oriented reward signal, enhancing the efficiency and success of ToD systems.
- To better and flexibly evaluate the efficiency and success rate of LLM-powered ToD systems, we introduce goal-driven user simulation based on GPT-4 to assess the performance of the ToD system.
- We conduct comprehensive experiments including automatic metrics evaluation, user simulator based assessment, case study and human evaluation, which fully validate the effectiveness of our model.

2 Related Work

ToD systems have been as essential tools for facilitating various tasks such as vairous bookings or reservations scenarios in natural language conversations. These systems aim to provide human-like interactions, making it convenient for users to engage with them seamlessly. In recent years, there has been significant progress in the development of ToD systems, with various approaches and techniques contributing to their enhancement.

Some of the earlier ToD models, such as those presented in the works of He et al. (2022), Lee (2021), Sun et al. (2023), and Wu et al. (2023a), primarily focused on generating responses based solely on the current dialogue context. While these models showed promise, they often lacked the ability to consider the broader context or incorporate dialogue states effectively. To address this limitation, researchers have explored policy optimization methods, as highlighted in the studies conducted by Wang et al. (2020a) and Wang et al. (2020b). These approaches leverage ground-truth dialogue states to inform the response generation process, thereby enabling more contextually relevant and accurate responses. Incorporating both text information and dialogue states has been another promising avenue for improving ToD systems. Lubis et al. (2020) and Lee (2021) are notable examples of research efforts that have successfully integrated these two aspects. This approach allows ToD systems to have a more comprehensive understanding of the conversation, enabling them to generate responses that are not only contextually appropriate but also take into account the underlying task objectives. Furthermore, reinforcement learning methods, as demonstrated in studies by Wu et al. (2023b), Bang et al. (2023), and Feng et al. (2023), have gained recognition for their effectiveness in enhancing ToD systems. These methods leverage feedback and rewards to fine-tune the dialogue generation process, resulting in responses that are not only context-aware but also optimized for specific task-oriented goals.

Recently, a new paradigm in natural language processing has emerged, characterized by the advent of sophisticated LLMs such as ChatGPT (OpenAI, 2021), GPT-4 (OpenAI, 2023), Llama2 (Touvron et al., 2023), Bard, and others. These models have significantly enhanced a variety of applications owing to their superior understanding and generation capabilities. The advancements these LLMs embody are markedly distinct, with performance that greatly surpasses that of earlier pretraining language models like BERT (Devlin et al., 2018), T5 (Raffel et al., 2020), and GPT-2 (Radford et al., 2019), among others.

In terms of the LLM-based ToD research, Madotto et al. (2020) assess the few-shot capability of language models in Natural Language Understanding, Dialogue State Tracking, Dialogue Policy and Natural Language Generation tasks. Hudeček and Dušek (2023) evaluate Instruction-finetuned LLMs' ability to complete multi-turn tasks and interact with external databases in the context of established task-oriented dialogue benchmarks. Snell et al. (2022) formulate goal-oriented dialogue as a partially observed Markov decision process, interpreting the language model as a representation of both the dynamics and the policy. Recently, Hu et al. (2023) propose a new framework to leverage LLM as the user simulator and utilize the feedback of this simulation to optimize the ToD model. Li et al. (2023) introduce a novel prompting framework called Directional Stimulus Prompting for guiding black-box LLMs toward desired output, which employ a small tunable policy model to generate the hint to guide the LLMs.

3 Methodology

3.1 Overview

For dialogue response generation, we consider an input dialogue history space denoted as $\mathbf{X}$, and a response output space referred to as $\mathbf{Y}$. LLMs have shown remarkable capabilities in generating responses by incorporating instructions that describe the task, a few demonstration examples, and the input dialogue history $\mathbf{x}$ within the prompt. However, there are challenges when it comes to steering LLMs towards desired outputs, particularly for achieving fine-grained, query-specific behaviors. In the context of ToD, different dialogue systems need to respond to user queries across various domains using actions such as informing, requesting, confirming, and providing domain-specific slot values. In such scenarios, solely relying on task-specific instructions and a handful of examples may not consistently yield satisfactory and relevant responses. Additionally, dealing with long-term memory and maintaining efficiency poses further challenges in LLM-based ToD systems.

To address these issues, we propose the incorporation of future dialogue action hints denoted as $\boldsymbol{z}$ into the prompt, inspired by the Directional Stimulus Prompting (DSP) approach (Li et al., 2023). These hints serve as guidance for achieving the desired response. For each input query, we generate these hints using a small, adaptable policy language model, $p_{POL}(\boldsymbol{z}|\boldsymbol{x})$. Subsequently, we combine the generated hint, $\boldsymbol{z}$, with the original dialogue history, $\boldsymbol{x}$, to construct the prompt that guides the LLM towards generating its output, represented as $p_{LLM}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z})$.

3.2 Anticipating Future Dialogue Actions

To anticipate future dialogue actions by the user, we train a policy model that predicts future dialogue

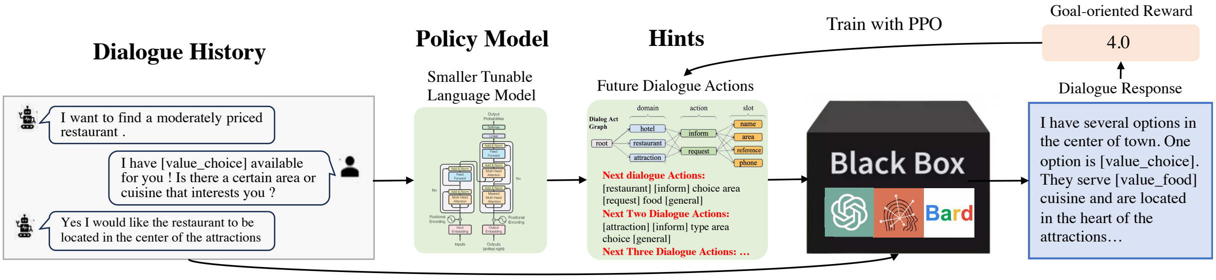

Figure 2: Model Architecture. The policy model is trained to generate future dialogue actions to induce the LLM to perform specific tasks, via supervised finetuning and PPO based reinforcement learning.

actions for LLMs, using supervised fine-tuning of a pre-trained LM such as T5 on a small collection of labeled data (1% or 10%).

To enhance the ability of LLMs to generate task-specific responses, we employ the anticipated future dialogue actions, spanning from the current turn until the end of the conversation, as contextual cues for guiding the LLM in generating responses to the queries from the current user turn. These cues are denoted as $\boldsymbol{z}$, which convey the anticipated future dialogue actions that the dialogue system should respond to. The resulting dataset, denoted as $\mathcal{D} = (\boldsymbol{x}, \boldsymbol{z})$, comprises pairs of dialogue histories and future action sequences. Specifically, given a dialogue history with n turns, represented as $\boldsymbol{x} = (x_1, x_2, x_3, \ldots, x_n)$, and corresponding predicted actions for each response turn, denoted as $\mathbf{a} = (a_1, a_2, a_3, \ldots, a_n)$, we formulate the future predicted actions for the i-th turn as $z_i = (a_i, a_{i+1}, \ldots, a_n)$. Subsequently, we fine-tune the policy model by maximizing the log-likelihood through the following objective:

$$\mathcal{L}_{\mathrm{POL}} = -\mathbb{E}_{(\boldsymbol{x},\boldsymbol{z})\sim\mathcal{D}} \log p_{\mathrm{TOD}} \left(\boldsymbol{z} \mid \boldsymbol{x}\right) \quad (1)$$

This framework enables our model to generate responses that align with the underlying dialogue actions, resulting in more contextually appropriate and task-specific outputs for the current user query. To better modify the hints toward achieving high success rate on the dialogue goal, we continue to incorporate reinforcement learning (RL) to further fine-tune the policy model based on the goal-oriented reward. The detailed approach is elaborated below.

3.3 Goal-oriented Reward

Our objective is to guide the generation of the LLM towards our desired target by optimizing an alignment measure denoted as $\mathcal{R}(\boldsymbol{x}, \boldsymbol{y})$, which measures whether the response $\boldsymbol{y}$ achieves predefined dialogue goals for the input of $\boldsymbol{x}$.

In each dialogue, the overall goal can be subdivided into several predefined sub-goals, denoted as $\mathbf{g} = (g_1, g_2, g_3, \ldots, g_m)$. The success of each sub-goal can be measured by assessing whether the ToD system provides the corresponding information. For instance, in the MultiWoZ dataset, these sub-goals encompass tasks such as supplying a reference ID, phone number, address, and so forth. Consequently, for each system response turn, we can calculate a turn-level goal reward that quantifies how many sub-goals it accomplishes. This can be mathematically formulated as follows:

$$r_i = \lambda \sum_{j=0}^{m} g_j \quad (2)$$

where r_i represents the reward for the i-th turn, and λ is a hyperparameter to scale the reward. The value of each g_j is determined by whether the ToD system provides the corresponding information: if it does, g_j is assigned a value of 1; otherwise, it is set to 0.

Meanwhile, given that the parameters of the black-box LLM are neither accessible nor adjustable, we resort to enhancing the policy model's optimization. This involves generating future dialogue actions as hints, which in turn direct the LLMs' generation process towards the maximization of our defined objective.

$$\mathcal{R}_{LLM}(\boldsymbol{x}, \boldsymbol{y}) = \mathcal{R}(\boldsymbol{x}, \boldsymbol{y}) \quad (3)$$

$$\boldsymbol{y} \sim p_{LLM}(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z}) \quad (4)$$

However, the optimization approach described above poses an intractable problem for the policy model. In order to tackle this challenge, we reframe the optimization of the policy model as a reinforcement learning (RL) problem and leverage the proximal policy optimization (PPO) algorithm (Schulman et al., 2017).

We use the policy model to initialize a policy network $\pi_0 = p_{POL}$ and then update π using PPO.

The process through which the policy model generates a sequence of future actions $\boldsymbol{z}$ can be viewed as an interaction in the context of RL, defined by the tuple $<\mathbf{S}, \mathbf{A}, \mathbf{r}, \mathbf{P}>$. Here, $\mathbf{S}$ represents the state space, $\mathbf{A}$ denotes the action space, $\mathbf{r}$ corresponds to the reward function, and $\mathbf{P}$ signifies the state-transition probability. In each interaction with the environment, the agent selects an action (token) based on the probability distribution defined by the current policy network $\pi(\mathbf{z}|\mathbf{x}, \mathbf{z} < \mathbf{t})$. The interaction process concludes when an end-of-sequence token is chosen, resulting in the generation of the entire sequence of future dialogue actions. The policy network π can be improved through fine-tuning, aiming to optimize the reward $\mathbf{r}$ associated with the RL framework.

To avoid the policy network π deviating too far from the initial policy model p_{POL}, we also introduce a KL-divergence penalty into the current reward function. Therefore, the final reward formula is:

$$r(\boldsymbol{x}, \boldsymbol{y}) = \mathcal{R}_{LLM}(\boldsymbol{x}, \boldsymbol{y}) - \beta \log \frac{\pi(\boldsymbol{y} \mid \boldsymbol{x})}{p_{\text{TOD}}(\boldsymbol{y} \mid \boldsymbol{x})} \tag{5}$$

3.4 Goal-driven User Simulation Assessment

As highlighted by (Wu et al., 2023a) in their study on the Inform and Success metrics, these have inherent issues. The evaluation procedure requires the model to generate placeholders, and the metrics consider whether the placeholders satisfy user goals. Consequently, a model that generates more placeholders can misleadingly appear to perform better. When a fixed response such as "[value_name] [value_phone] [value_address] [value_postcode] [value_reference] [value_id]" is consistently used for every turn during evaluation with the standardized evaluation script, it yields state-of-the-art results in terms of Inform and Success scores when compared to baseline models.

Therefore, these problems necessitate a new method to assess the performance of LLM-based ToD systems. Given a dialogue goal $\boldsymbol{g}$, we design a suitable prompt to enable GPT-4 to act as the user to propose the user queries $x^* = \{x_1^*, x_2^*, x_3^* \dots x_n^*\}$, for which the LLM-based ToD's responses are $y^* = \{y_1^*, y_2^*, y_3^* \dots y_n^*\}$ in the turn-by-turn interaction. In these simulations, the efficiency can be calculated as the average number of turns. Due to the strong understanding of GPT-4, the success will be assessed by GPT-4 again according to the dialogue goal and this simulated dialogue.

3.5 Implementation

We employ the T5 model (base version) (Raffel et al., 2020) as our policy model. Concurrently, GPT-3.5-turbo (OpenAI, 2021) serves as the specific LLM.

Our process commences with the supervised fine-tuning of the T5 model, centering on the future dialogue actions prediction task. Post the initial phase, we augment the T5's capabilities by introducing a goal-oriented reward system and the NLPO method (NLPO (Ramamurthy et al., 2022) is one of the PPO Algorithms, tailored for language generators). These enhancements are geared towards optimizing the model through reinforcement learning, aligning it closely with user goal completion.

To adapt to the distinct characteristics of dataset annotation and evaluation metrics, we have tailored the sub-goals to specifically provide 'PHONE', 'ADDRESS', 'POST', and 'REFERENCE ID'. Even with this specific design, the framework of our goal-oriented reward system retains its flexibility, allowing for extensions and adaptations to other ToD systems through tailored goal amendments. In this context, we have set the scaling reward parameter λ at 3 to achieve a balanced optimization.

4 Experiments

4.1 Dataset and Evaluation Metrics

MultiWoZ 2.1 (Eric et al., 2020) is the improved version of MultiWOZ 2.0 (Budzianowski et al., 2018) which is a released multi-domain dialogue dataset spanning 7 distinct domains and containing over 10,000 dialogues. Moreover, MultiWOZ 2.1 also includes user dialogue acts as well as multiple slot descriptions per dialogue state slot.

Schema Guided Dialogue (SGD) (Rastogi et al., 2020) consists of schemas outlining the interface of different APIs and annotated dialogues, including over 16k multi-domain conversations spanning 16 domains.

Inform and Success are the metrics related to dialogue task completion - whether the system provides an appropriate entity (Inform rate) and answers all the requested attributes (Success rate).

Model	Training Data	MultiWoZ 2.1 Inform	MultiWoZ 2.1 Success	SGD Inform	SGD Success
Standard Prompting	-	72.8	44.2	64.3	57.2
DSP w/ SFT (Li et al., 2023)	1%	76.0	64.3	71.1*	69.3*
DSP (Li et al., 2023)	1%	87.3	78.7	79.6*	75.2*
ProToD-	1%	90.4	80.1	84.3	77.5
ProToD	1%	94.3	82.7	87.9	82.1
DSP w/ SFT (Li et al., 2023)	10%	75.0	67.7	73.7*	69.9*
DSP (Li et al., 2023)	10%	95.0	84.0	82.0*	78.5*
ProToD-	10%	95.3	85.0	88.1	80.9
ProToD	10%	**96.2**	85.8	**90.6**	**84.2**
SimpleTOD (Hosseini-Asl et al., 2020)	100%	85.0	70.5	-	-
DoTS (Jeon and Lee, 2021)	100%	86.7	74.2	-	-
PPTOD (Su et al., 2021)	100%	87.1	79.1	-	-
UBAR (Yang et al., 2021)	100%	95.7	81.8	-	-
GALAXY (He et al., 2022)	100%	95.3	**86.2**	-	-

Table 1: Comparison of response generation performance across different methods. The results with $*$ mean the reimplementation results by us. The best results are highlighted in bold, while the top performance in each category is underscored with an underline. ProToD- is the ablation setting which still incorporates future dialogue actions but omits rewards based on task-success, using only BLEU as rewards.

4.2 Baselines

Standard Prompting: We design instructions to let LLMs reply to the previous dialogue history.

DSP (Li et al., 2023): In Directional Stimulus Prompting (DSP), "directional stimulus" is introduced into the prompt to provide more precise guidance for LLMs. This stimulus acts as a cue to guide LLMs in generating desired outputs. A small tunable model, such as T5, is used to create this stimulus for each query, allowing optimization of LLMs through a smaller policy model. This policy model is trained through supervised fine-tuning with labeled data and reinforcement learning using rewards, aiming to align LLM behavior with desired outcomes.

DSP w/ SFT represents the ablation study in DSP, performing the supervised fine-tuning (SFT) on the pre-trained LM, without the further reinforcement learning training.

SimpleTOD (Hosseini-Asl et al., 2020) adopts a unified approach, treating all these sub-tasks as a single sequence prediction problem, leveraging pre-trained, open-domain, causal language models like GPT-2 as its base model.

DoTS (Jeon and Lee, 2021) is a task-oriented dialogue system that uses a simplified input context instead of the entire dialogue history. To address the loss of contextual information from previous conversational turns. DoTS tracks the domain state in addition to the belief state and uses it for the input context.

PPTOD (Su et al., 2021) proposes a unified solution that overcomes the limitations of the traditional cascaded generation approach. Its integrated architecture efficiently minimizes error accumulation and data annotation overheads. It adopts a multi-task pre-training strategy to leverage insights from diverse dialogue corpora.

UBAR (Yang et al., 2021) is a ToD system that models entire dialog sessions by fine-tuning GPT-2 on sequences encompassing user input, belief states, system actions, and responses.

GALAXY (He et al., 2022): This method is the previous end-to-end fully supervised training SOTA model. The Galaxy model is a pre-trained conversational system that acquires dialog strategies by leveraging a combination of limited labeled dialog data and extensive unlabeled dialog datasets through a semi-supervised learning approach. They incorporate a task to predict dialog actions as a means of improving dialog policy during the pre-training phase and utilize a consistency regularization component to enhance the acquired representations with the aid of unlabeled dialog data.

4.3 ProToD Performance

We assessed the efficacy of our ProToD approach by evaluating it on GPT-3.5-turbo and comparing its performance with the LLM-guided DSP work (Li et al., 2023), and other prominent task-oriented dialogue models like SimpleTOD (Hosseini-Asl et al., 2020), DoTS (Jeon and Lee, 2021), PPTOD (Su et al., 2021), UBAR (Yang et al., 2021), and GALAXY (He et al., 2022). These models were trained on the comprehensive training set consisting of 8438 dialogues. The comparative analysis of their overall performance is summarized in Table 1.

Table 1 illustrates that ProToD consistently eclipses the DSP model in performance, a trend observed when trained with both 1% and 10% of the data. ProToD, when trained with only 1% of the data, results in an average improvement of 6.6% over the DSP model. These findings underscore the importance of anticipating future actions for LLM-guided ToD.

In scenarios where only 10% of the training data in MultiWoZ 2.1 is utilized, ProToD further outperforms fully supervised, end-to-end training models UBAR and GALAXY, as evidenced by the Inform metric. Even with the constrained training dataset of 1%, ProToD outperforms SimpleTOD, DoTS, and PPTOD in both the Inform and Success metrics.

The superior performance of ProToD can be attributed to its utilization of anticipated future actions along with the LLMs' effective understanding and generation capabilities. With just the hints provided by the T5 model, ProToD outperforms intricate models that are grounded in complex architecture and tailored dialogue features. Noteworthy is ProToD's exemplary performance when trained on smaller datasets, a significant advantage especially when dealing with dialogues where dataset annotation is particularly costly.

4.4 Ablation Study

We perform an ablation study utilizing ProToD**-**, which still incorporates future dialogue actions but omits rewards based on task-success, using only BLEU as reward. Even without the goal-oriented reward, ProToD**-** still outperforms DSP, a difference that becomes more pronounced when training is conducted using only 1% of the standard data volume. This underscores the module's inherent effectiveness, as evidenced by its positive contributions to both Inform and Success metrics. Furthermore, a comparative analysis between ProToD and ProToD- further illuminates the significant enhancements afforded by the incorporation of the goal-oriented award module.

4.5 Goal-driven User Simulation Assessment

We randomly sample 100 dialogues and employ GPT-4 as the user simulator to conduct the dialogue simulation. Then, we calculate the efficiency and let GPT-4 to assess whether this dialogue completed the dialogue goal and how satisfactory the user feels. In terms of evaluation metrics, Success Rate (SU) considers whether the dialogue fulfills the goal, Efficiency means the average turns needed to complete the goal (if this dialogue fails to complete the goal, we use the maximum turns of 10). Additionally, we adopt the previous user satisfaction work (Sun et al., 2021) to set the satisfaction score (SA ranging from 1-5 and prompt GPT-4 to provide this satisfaction score.

As illustrated in Table 2, ProToD surpasses DSP with a success rate of 69.2% against DSP's 61.5%. This indicates ProToD's enhanced capability to achieve dialogue goals. When considering efficiency, ProToD's score of 5.7 outperforms DSP's 6.3, suggesting that ProToD requires fewer interactions for task completion. Furthermore, the higher satisfaction score of 4.3 for ProToD, compared to DSP's 4.0, implies that users may find interactions with ProToD more intuitive and satisfying. This holistic improvement highlights ProToD's effectiveness in multiple aspects.

As previously discussed, the current evaluation metrics fall short in assessing the semantic quality of responses and their effectiveness in task completion. These limitations underscore a critical gap in the evaluation paradigm for ToD systems. Our proposed method, in conjunction with the three metrics outlined earlier, presents a transformative approach to address this challenge. By integrating these advanced measures, we aim to offer a more comprehensive and nuanced evaluation framework, ensuring that both the user satisfaction and task accomplishment efficacy of the dialogue systems are meticulously assessed and optimized.

Method	SU↑	EF↓	SA↑
DSP (Li et al., 2023)	61.5	6.3	4.0
ProToD	69.2	5.7	4.3

Table 2: Evaluation performance of dialogue response generation through simulation by GPT-4, where SU, EF and SA denote as Success Rate, Efficiency and Satisfaction respectively

Conversation Goal:
You are planning your trip in Cambridge. You are looking for a **restaurant**. The restaurant should be in the **centre** and should be in the **moderate** price range. Make sure you get **food type** and **postcode**. You are also looking for **places to go** in town. The attraction should be in the type of **mutliple sports** and should be **in the same area as the restaurant**. If there is no such attraction, how about one that is in **the type of architecture**. Make sure you get **phone number**, **entrance fee**, and **address**

Conversation History:
User: I want to find a moderately priced restaurant .
Assistant: I have [value_choice] available for you ! Is there a certain area or cuisine that interests you ?
User: Yes I would like the restaurant to be located in the center of the attractions .

DSP Response: There are [value_choice] restaurants in the center of the attractions . what type of food are you interested in ?

ProToD Response: I have several options in the center of town. One option is [value_choice]. They serve [value_food] cuisine and are located in the heart of the attractions. Another option is [value_choice]. They have a variety of cuisines and are also centrally located. Would you like me to provide you with their phone?

4.6 Case Study

In our study, we employ a case study approach to further bolster our motivation. This case study comprises four key components: the conversation goal, history, DSP model response, and ProToD model response. When we examine the ProToD response in comparison to the DSP response, it becomes evident that the ProToD model offers two notable advantages. Firstly, the ProToD response excels in providing current options and strives to meet the user's objectives as efficiently as possible, rather than simply asking for the type of food, thereby significantly enhancing dialogue efficiency. For each option presented, the ProToD-generated response includes comprehensive information encompassing cuisine types and location clarifications. Secondly, the ProToD response addresses the user's intent confirmation by providing a phone number when it is required, potentially increasing the success rate of the dialogue. Together, these two strengths of the ProToD model – enhancing the detail and relevance of information provided, and effectively confirming and fulfilling user intents – solidify its role in significantly improving both the Inform and Success metrics within the MultiWoZ 2.1 dataset. This underlines ProToD's effectiveness in delivering more efficient, successful, and user-aligned dialogue experiences.

5 Conclusion

In this study, we present the **ProToD** model, an enhancement of the LLM-powered ToD system that incorporates future dialogue action anticipation and goal-oriented reward motivation. By utilizing future actions as cues to guide LLMs, our model offers more comprehensive responses and enhances the efficiency of dialogues. The integration of goal-oriented rewards further fine-tunes the cues for LLMs, resulting in improved dialogue task completion rates through a reinforcement learning framework. Additionally, we introduce a goal-driven user simulation assessment based on GPT-4, providing a novel perspective to better evaluate dialogue efficiency and user satisfaction levels. Our validation process assesses the effectiveness of ProToD by examining performance enhancements in Inform and Success metrics using the MultiWoZ 2.1 and SGD dataset. Furthermore, we present case studies and user simulation assessments that illustrate the improvements in dialogue efficiency and user satisfaction achieved by our model.

6 Limitations

Firstly, the MultiWoZ dataset is currently unrivaled in its scale and comprehensiveness, equipped with dialogue goals and sub-goals for assessing dialogue task completion. However, to refine and broaden the evaluation landscape and our method's generalization, we require additional datasets to effectively evaluate and refine our method.

Additionally, our approach is still reliant on LLMs, and the understanding and generation capabilities can vary among different models. While the DSP work (Li et al., 2023) has compared performance with CodeX, there is a need to evaluate more recent and superior LLMs like GPT-4, Llama 2, and others within this framework.

References

Namo Bang, Jeehyun Lee, and Myoung-Wan Koo. 2023. Task-optimized adapters for an end-to-end task-oriented dialogue system. *arXiv preprint arXiv:2305.02468*.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Mihail Eric, Rahul Goel, Shachi Paul, Abhishek Sethi, Sanchit Agarwal, Shuyang Gao, Adarsh Kumar, Anuj Goyal, Peter Ku, and Dilek Hakkani-Tur. 2020. MultiWOZ 2.1: A consolidated multi-domain dialogue dataset with state corrections and state tracking baselines. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 422–428, Marseille, France. European Language Resources Association.

Yihao Feng, Shentao Yang, Shujian Zhang, Jianguo Zhang, Caiming Xiong, Mingyuan Zhou, and Huan Wang. 2023. Fantastic rewards and how to tame them: A case study on reward learning for task-oriented dialogue systems. *arXiv preprint arXiv:2302.10342*.

Adam M Grant and Susan J Ashford. 2008. The dynamics of proactivity at work. *Research in organizational behavior*, 28:3–34.

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. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 10749–10757.

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.

Zhiyuan Hu, Yue Feng, Anh Tuan Luu, Bryan Hooi, and Aldo Lipani. 2023. Unlocking the potential of user feedback: Leveraging large language model as user simulator to enhance dialogue system. *arXiv preprint arXiv:2306.09821*.

Vojtěch Hudeček and Ondřej Dušek. 2023. Are llms all you need for task-oriented dialogue? *arXiv preprint arXiv:2304.06556*.

Hyunmin Jeon and Gary Geunbae Lee. 2021. Domain state tracking for a simplified dialogue system. *arXiv preprint arXiv:2103.06648*.

Yohan Lee. 2021. Improving end-to-end task-oriented dialog system with a simple auxiliary task. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 1296–1303.

Zekun Li, Baolin Peng, Pengcheng He, Michel Galley, Jianfeng Gao, and Xifeng Yan. 2023. Guiding large language models via directional stimulus prompting. *arXiv preprint arXiv:2302.11520*.

Nurul Lubis, Christian Geishauser, Michael Heck, Hsien-chin Lin, Marco Moresi, Carel van Niekerk, and Milica Gašić. 2020. Lava: Latent action spaces via variational auto-encoding for dialogue policy optimization. *arXiv preprint arXiv:2011.09378*.

Andrea Madotto, Zihan Liu, Zhaojiang Lin, and Pascale Fung. 2020. Language models as few-shot learner for task-oriented dialogue systems. *arXiv preprint arXiv:2008.06239*.

OpenAI. 2021. ChatGPT. `https://www.openai.com/gpt-3/`. Accessed: April 21, 2023.

OpenAI. 2023. Gpt-4 technical report.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Rajkumar Ramamurthy, Prithviraj Ammanabrolu, Kianté Brantley, Jack Hessel, Rafet Sifa, Christian Bauckhage, Hannaneh Hajishirzi, and Yejin Choi.

2022. Is reinforcement learning (not) for natural language processing?: Benchmarks, baselines, and building blocks for natural language policy optimization. *arXiv preprint arXiv:2210.01241*.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 8689–8696.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Charlie Snell, Mengjiao Yang, Justin Fu, Yi Su, and Sergey Levine. 2022. Context-aware language modeling for goal-oriented dialogue systems. *arXiv preprint arXiv:2204.10198*.

Yixuan Su, Lei Shu, Elman Mansimov, Arshit Gupta, Deng Cai, Yi-An Lai, and Yi Zhang. 2021. Multi-task pre-training for plug-and-play task-oriented dialogue system. *arXiv preprint arXiv:2109.14739*.

Haipeng Sun, Junwei Bao, Youzheng Wu, and Xiaodong He. 2023. Mars: Modeling context & state representations with contrastive learning for end-to-end task-oriented dialog. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11139–11160.

Weiwei Sun, Shuo Zhang, Krisztian Balog, Zhaochun Ren, Pengjie Ren, Zhumin Chen, and Maarten de Rijke. 2021. Simulating user satisfaction for the evaluation of task-oriented dialogue systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2499–2506.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Jianhong Wang, Yuan Zhang, Tae-Kyun Kim, and Yunjie Gu. 2020a. Modelling hierarchical structure between dialogue policy and natural language generator with option framework for task-oriented dialogue system. *arXiv preprint arXiv:2006.06814*.

Kai Wang, Junfeng Tian, Rui Wang, Xiaojun Quan, and Jianxing Yu. 2020b. Multi-domain dialogue acts and response co-generation. *arXiv preprint arXiv:2004.12363*.

Qingyang Wu, Deema Alnuhait, Derek Chen, and Zhou Yu. 2023a. Using textual interface to align external knowledge for end-to-end task-oriented dialogue systems. *arXiv preprint arXiv:2305.13710*.

Qingyang Wu, James Gung, Raphael Shu, and Yi Zhang. 2023b. Diacttod: Learning generalizable latent dialogue acts for controllable task-oriented dialogue systems. *arXiv preprint arXiv:2308.00878*.

Yunyi Yang, Yunhao Li, and Xiaojun Quan. 2021. Ubar: Towards fully end-to-end task-oriented dialog system with gpt-2. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14230–14238.