Roll Up Your Sleeves: Working with a Collaborative and Engaging Task-Oriented Dialogue System

Lingbo Mo, Shijie Chen*, Ziru Chen*, Xiang Deng†, Ashley Lewis†, Sunit Singh†
Samuel Stevens†, Chang-You Tai†, Zhen Wang†, Xiang Yue†, Tianshu Zhang†, Yu Su‡, Huan Sun‡
The Ohio State University
{mo.169, chen.10216, chen.8336, deng.595, lewis.2799, singh.1790,
stevens.994, tai.97, wang.9215, yue.149, zhang.11535, su.809, sun.397}@osu.edu

Abstract

We introduce TACOBOT, a user-centered task-oriented digital assistant designed to guide users through complex real-world tasks with multiple steps. Covering a wide range of cooking and how-to tasks, we aim to deliver a collaborative and engaging dialogue experience. Equipped with language understanding, dialogue management, and response generation components supported by a robust search engine, TACOBOT ensures efficient task assistance. To enhance the dialogue experience, we explore a series of data augmentation strategies using LLMs to train advanced neural models continuously. TACOBOT builds upon our successful participation in the inaugural Alexa Prize TaskBot Challenge, where our team secured third place among ten competing teams. We offer TACOBOT as an open-source framework that serves as a practical example for deploying task-oriented dialogue systems.[1]

1 Introduction

Task-Oriented Dialogue (TOD) systems have shown promise in achieving user goals through conversational interactions. However, existing TOD systems focus on users providing information while the system performs tasks. In contrast, our task bot assists users in executing tasks themselves by providing accurate information and guidance.

However, we face several challenges mainly in two folds: (1) Existing TOD systems prioritize functional goals at the expense of user experience. (2) Inadequate in-domain training data, as modern neural models require large amounts of data, and acquiring annotations through crowdsourcing is costly. In this paper, we present TACOBOT, a task-oriented dialogue system designed to assist users in completing multi-step cooking and how-to tasks. Built upon our previous bot (Chen et al., 2022) deployed in the Alexa Prize TaskBot Challenge, TACOBOT aims to deliver a collaborative and engaging user experience. Figure 1 showcases a partial example dialogue.

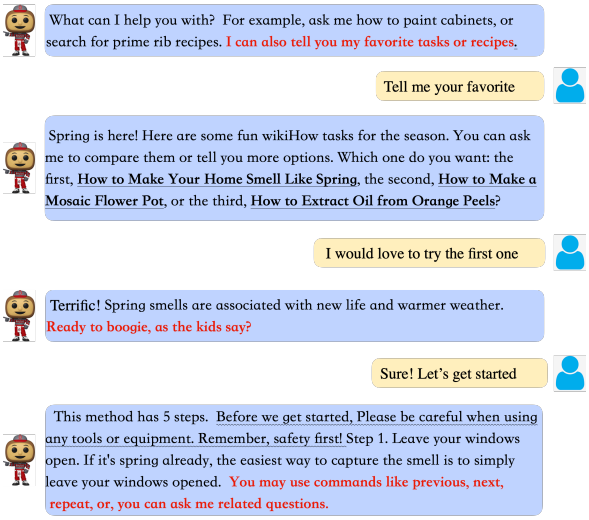

Figure 1: An example dialogue showing first few turns.

Our contributions include: (1) Developing a modularized TOD framework with accurate language understanding, flexible dialogue management, and engaging response generation. (2) Exploring data augmentation strategies, such as leveraging GPT-3 to synthesize large-scale training data. (3) Introducing clarifying questions about nutrition for cooking tasks to personalize search and better cater to user needs. (4) Incorporating chit-chat functionality, allowing users to discuss open topics of interest beyond the task at hand.

2 System Design

2.1 System Overview

TACOBOT follows a canonical pipeline approach for TOD systems. The system consists of three main modules: Natural Language Understanding (NLU), Dialogue Management (DM), and Response Generation (RG). NLU module preprocesses the user's utterance to determine their intent.

* Team co-leads in the challenge with equal contribution.
† Other authors in alphabetical order. ‡ Faculty advisors.
[1] Code and datasets will be made available upon acceptance.

DM module, designed with a hierarchical finite state machine, controls the dialogue flow, handles exceptions, and guides the conversation towards task completion. RG module generates responses using relevant knowledge and additional modalities to enhance user engagement. Each module is supported by a well-organized knowledge backend and search engine, capable of connecting with various sources to provide optimal user assistance.

2.2 Natural Language Understanding

Our bot employs a robust NLU pipeline, combining pre-trained language models and rule-based approaches. The key component is *Intent Recognition*, where we define ten user intents under four categories to support diverse user initiatives. Recognizing that real user utterances can involve multiple intents, we address intent recognition as a multi-label classification problem and filter model predictions based on dialogue states. Additionally, certain intents, like **Navigation**, are further parsed using regular expressions for finer granularity.

To develop a high-quality multi-label classification model despite limited data, we employ data augmentation and domain adaptation techniques. We leverage existing datasets (Rastogi et al., 2019) for common intents like **Sentiment** and **Question**, while utilizing the in-context learning capability of GPT-3 for other intents. By synthesizing initial utterances with intent descriptions and few-shot examples, we create a foundation for training data. To expand the dataset, we transform synthetic utterances into templates, substituting slot values with placeholders and filling them with sampled values to generate actual training utterances. Additionally, we incorporate linguistic rules, neural paraphrase models, and user noise, such as filler words, to enhance data diversity and improve the robustness of our intent recognition module.

2.3 Dialogue Management

We design a hierarchical finite state machine for the DM component, consisting of three phases: Task Search, Task Preparation, and Task Execution. Each phase comprises multiple fine-grained dialogue states, as depicted in Figure 2.

In the **Task Search phase**, users can search for how-to tasks or recipes directly by issuing a query or ask for task recommendations. TacoBot retrieves search results from the backend search engine (Section 2.4) and presents candidate tasks for users to compare and select. Once users choose an

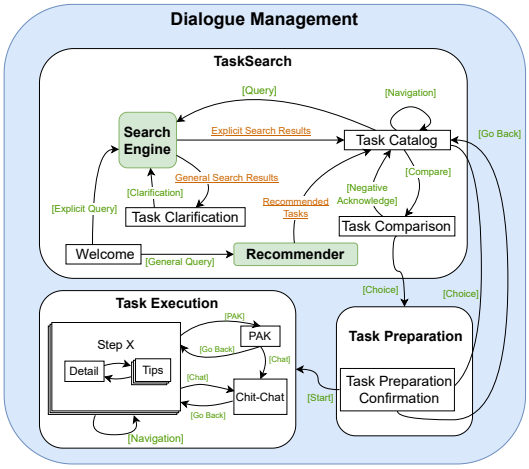

Figure 2: Dialogue Management Diagram. White boxes represent dialogue states and green boxes represent supporting modules. Bidirectional edges represent reflexive transitions. Green texts represent user intent and orange texts denote search engine output.

option, they enter the **Task Preparation phase**. In this phase, users review detailed information about the selected task and decide whether to proceed or search for another task. If users change their mind, they can go back to Task Search and find an alternative task. If they commit to the chosen task, they proceed to the **Task Execution phase**. During this last phase, users follow the step-by-step instructions provided by TacoBot to complete the task. The utility module, such as the QA module, assists users throughout this phase. Each step of the task has its own state, and for how-to tasks, we break down lengthy steps into shorter instructions, details, and tips for better user comprehension.

DM performs state transitions and selects response generators (Section 2.5) based on user input. The hierarchical design of dialogue states allows for extensible and flexible transitions at different levels. A dialogue state history stack is maintained to facilitate easy navigation to previous states. User intents that do not trigger valid transitions provide contextualized help information to guide users through the dialogue. These design choices ensure stable yet flexible dialogue experiences for users.

2.4 Search Engine

TacoBot can support diverse tasks backed by large-scale corpus. For the cooking domain, we build a recipe corpus which contains 1.02M recipes based on Recipe1M+ dataset (Marın et al., 2019). Meanwhile, we build a wikiHow corpus that includes 93.1K how-to tasks collected from wikiHow

website[2]. On top of that, we construct a search engine for both domains based on Elastic search.

2.4.1 Ranking Strategy

To improve the relevance of search results and mitigate the issue of lexical similarity in Elastic search, we employ a query expansion technique that expands user queries by incorporating related words from task names, such as lemmatized verbs, nouns, and decomposed compound nouns. Additionally, we enhance search performance by implementing a neural re-ranking model based on BERT. This model assigns a score to each task by considering the task request and retrieved task titles as input. Training the re-ranker involves employing a weakly-supervised list-wise ranking loss and utilizing synthesized task queries via GPT-3 query simulation. We also propose the collection of weak supervision signals from Google's search engine to avoid the need for human annotation.

2.4.2 Personalized Search

In addition to implementing ranking strategies for accurate search results, our goal is to introduce personalization into the search engine, ensuring a better match with users' needs. To achieve this, we propose a method of asking clarifying questions during recipe searches and collaborating with users to understand their preferences regarding nutrition. Specifically, when a user provides a cooking task of interest, we actively engage in clarifying discussions with them about the desired level of nutrition in terms of *sugar*, *fat*, *saturates*, and *salt*, using the traffic lights definition established by the Food Standards Agency (FSA).

2.5 Response Generation

Our response generation module blends both infilling-based methods and neural models. On the one hand, we leverage handcrafted conditional rules to organize curated templates and their composition strategy according to the high-level states in our hierarchical finite-state machine. On the other hand, we build question answering (QA) models, a crucial functionality of TOD systems, to answer different questions from the user.

2.5.1 Question Type Classifier

Our QA system encompasses various question types, including in-context machine reading comprehension (MRC) for context-dependent questions, out-of-context (OOC) QA for open domain questions, frequently-asked questions (FAQ) retrieval for how-to tasks, and rule-based Ingredient and Substitute QA for cooking tasks.

Then, we develop a question type classifier that categorizes user questions into five types (MRC, OOC, FAQ, Ingredient, Substitute) for cooking tasks, and three types (MRC, OOC, FAQ) for how-to tasks. To improve classification accuracy, we concatenate the instruction of the current step (if available) as context with the input question. This combined sequence is then fed into a Roberta-base classifier. Our training set consists of 5,000 questions for each question type, allowing for effective differentiation between different types of questions.

2.5.2 Context-Dependent QA

We begin by annotating an in-context QA dataset comprising 5,183 QA pairs, out of which 752 are unanswerable questions. To ensure reliable responses, we employ Roberta-base to build an extractive QA model in two stages. Initially, we pre-train our model on SQuAD 2.0, followed by fine-tuning on our annotated QA dataset. Recognizing that users may inquire about previously shown steps, we enhance the context by concatenating the current step with the preceding n steps ($n = 2$) during both training and inference processes to prevent information gaps and hallucination.

2.5.3 Context-Independent QA

TACOBOT supports both in-context and context-independent questions. For **out-of-context QA**, we utilize FLAN-T5-XXL (Chung et al., 2022), an instruction-finetuned language model with 11B parameters. Under the zero-shot prompting setup, our bot is equipped to handle open-domain QA and demonstrate commonsense reasoning.

Additionally, our **FAQ** module leverages frequently-asked questions from the Community Q&A section of wikiHow articles, providing reliable answers sourced from real user questions and human expert responses. We use a retrieval module based on cosine similarity with question embeddings generated by a sentence-BERT encoder.

To address **ingredient-related queries**, we extract mentioned ingredients using a high-recall string matching mechanism against the recipe's ingredient list. If users lack a specific ingredient, TACOBOT can suggest substitutions when available, utilizing a dataset we collected covering 200 frequently used ingredients.

[2] https://www.wikihow.com

2.6 User Engagement

We develop several strategies to pursue an engaging dialogue experience in the following sections.

2.6.1 Chit-Chat

In real-world conversations, users often desire casual talk alongside the task. To enhance the user experience, TACOBOT offers chit-chat functionality, enabling flexible and diverse conversations. Inspired by Chirpy Cardinal (Chi et al., 2022), we integrate a chit-chat module into our TOD system. A template-based strategy is employed to identify user intent when entering and exiting chit-chat. The chit-chat process consists of three components.

Firstly, **Entity Tracker** monitors entities throughout the conversation, aligning responses with user intentions and focusing on the current topic. Recognized entities allow TACOBOT to access web sources (Wikipedia and Google) and provide intriguing information.

Secondly, **Chit-Chat Response Generator** incorporates various response generators: Neural Chat, Categories, Food, Aliens, Wiki, and Transition. Neural Chat uses BlenderBot-3B (Roller et al., 2021) to generate open-domain responses. Categories and Food generators elicit entity-related responses using templates. Transition facilitates smooth shifts between entities. Wiki enables users to discover engaging information in a conversational style. Aliens presents a five-part monologue series on extraterrestrial existence.

Lastly, **Intent Identification Model** determines if the user wants to continue or shift topics. TACOBOT proactively prompts users to return to the task after some chit-chat. Achieving natural transitions between chit-chat and task-oriented dialogue requires ongoing efforts.

2.6.2 People Also Ask

Furthermore, TACOBOT aims to enhance the dialogue experience by delivering captivating content. We leverage Google's "People Also Ask" (PAK) feature, which provides a list of related questions and summarized answers from web pages. This feature reveals popular topics of interest. To collect PAK data, we extract 30k common keywords from task titles in our recipe and wikiHow corpus, resulting in a total of 494k PAK QA pairs.

During task execution, PAK is presented as additional information. To avoid disrupting user focus, we limit the display frequency, currently showing it every 3 steps. Instead of directly displaying the PAK QA pair, we offer an interactive experience by presenting the question first, allowing users to decide if they want to view the corresponding answer. We also provide the option for users to engage in chit-chat if they choose to view PAK.

3 Conclusion

In this paper, we introduce TACOBOT, a modular task-oriented dialogue system designed to support users in accomplishing intricate daily tasks. We propose a comprehensive set of modules and approaches that contribute to the creation of a collaborative and engaging task bot. To establish a robust foundation for the entire system, we employ a series of data augmentation techniques leveraging LLMs. Additionally, we plan to make the framework and datasets open source, aiming to provide an example resource and inspire future endeavors in enhancing user-bot collaboration.

References

Shijie Chen, Ziru Chen, Xiang Deng, Ashley Lewis, Lingbo Mo, Samuel Stevens, Zhen Wang, Xiang Yue, Tianshu Zhang, Yu Su, et al. 2022. Bootstrapping a user-centered task-oriented dialogue system. *arXiv preprint arXiv:2207.05223*.

Ethan A Chi, Ashwin Paranjape, Abigail See, Caleb Chiam, Kathleen Kenealy, Swee Kiat Lim, Amelia Hardy, Chetanya Rastogi, Haojun Li, Alexander Iyabor, et al. 2022. Neural generation meets real people: Building a social, informative open-domain dialogue agent. *arXiv preprint arXiv:2207.12021*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Javier Marın, Aritro Biswas, Ferda Ofli, Nicholas Hynes, Amaia Salvador, Yusuf Aytar, Ingmar Weber, and Antonio Torralba. 2019. Recipe1m+: A dataset for learning cross-modal embeddings for cooking recipes and food images.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2019. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. *CoRR*, abs/1909.05855.

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Eric Michael Smith, Y-Lan Boureau, et al. 2021. Recipes for building an open-domain chatbot. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 300–325.