Conversational Task Agent

Jaime Russo, João Lopes, José Romano
NOVA School of Science and Technology, Caparica
{ja.russo,j.romano,jpdu.lopes}@campus.fct.unl.pt

Abstract

In this report, we outline the objectives, challenges, and scientific advancements of our team, in the development of a TaskBot for the Cooking Assistance Project 2024. Our overarching goal is to craft a TaskBot that embodies the qualities of being *helpful*, *multimodal*, *knowledgeable*, and *engaging*, offering users comprehensive guidance through complex cooking procedures. To realize this vision, we address three primary research inquiries: (1) Crafting Human-like Conversations, ensuring informative and knowledgeable interactions; (2) Leveraging Multimodal Capabilities, integrating voice, images, and videos to enrich user experience; and (3) Enhancing Adaptability with Zero-shot Conversational Flows, bolstering the TaskBot's resilience in navigating novel scenarios. Our TaskBot boasts a versatile repertoire, featuring innovative functionalities like adaptive recipe suggestions, voice-enabled video browsing, and the robust Cooking-Large Language Model (C-LLM), specifically trained to engage in detailed cooking-related dialogues. Based on user ratings and feedback, our observations affirm that the TaskBot excels in effectiveness and reliability, adeptly guiding users through cooking processes while seamlessly incorporating various multimodal stimuli.

1 Introduction

This report details the development of a conversational cooking assistant. This assistant, designed to empower users in the kitchen, tackles multi-step recipes. It retrieves relevant recipes, adapts instructions by considering available ingredients, and guides users through the cooking process through conversation.

The project is divided into three phases:

- **Phase 1: Task Retriever:** - In this phase, we implemented a search engine using the OpenSearch framework to enable users to find relevant tasks through natural language queries. We explored both text-based and embedding-based search functionalities.
- **Phase 2: Large Language and Vision Models:** - This phase focused on incorporating visual information processing. We implemented a CLIP encoder to enable cross-modal retrieval between text and image queries. Additionally, we integrated PlanLLM and ViLT large language models to answer user questions about the selected task and its visual aspects.
- **Phase 3: Dialog Manager:** - In the final phase, we built a dialog manager that leverages the capabilities developed in the previous phases. The dialog manager utilizes an intent detector to understand user goals and guides them through the chosen task using retrieved information and visual representations.

This report outlines the algorithms and implementation details for each phase, along with the evaluation methodologies and a critical discussion of the achieved results.

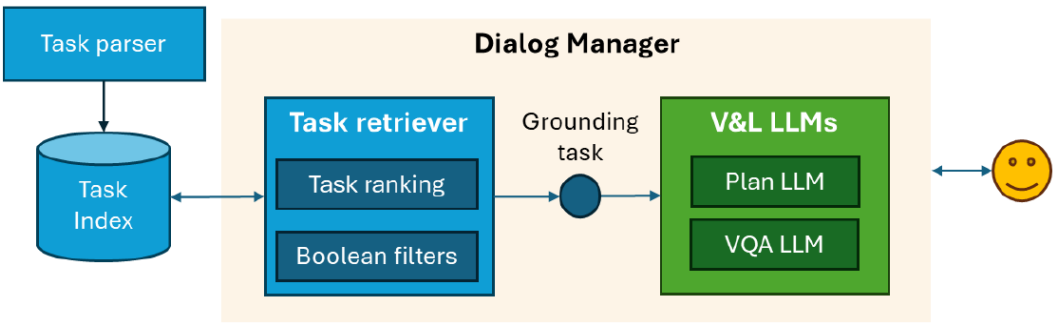

Figure 1: Architecture diagram of the project.

2 Algorithms and Implementation

In developing the Conversational Task Agent, a diverse array of algorithms has been leveraged, each strategically employed to address the multifaceted needs of users engaging in complex manual tasks. These algorithms serve as the backbone, orchestrating the interactions between the user and the system, ensuring a tailored and adaptive experience.

The **Term-based Search algorithm**, for instance, acts as the initial gateway, retrieving relevant tasks based on user input, be it a query related to ingredients, task titles, or descriptions. This enables users to effortlessly explore a wide spectrum of tasks, from baking recipes to DIY projects, simply by expressing their intent in natural language.

Meanwhile, the **Vector Embedding with KNN algorithm** takes the user experience to the next level by delving into the realm of semantic similarity. By representing tasks in a continuous vector space and employing K-nearest neighbors (KNN) for similarity calculations, the system can recommend tasks that closely align with the user's preferences and previous interactions. This functionality proves invaluable, especially when users seek variations or alternatives to a specific task, such as exploring similar recipes or projects with overlapping ingredients or steps.

Moreover, the Boolean Query for Ingredient Matching algorithm addresses a common challenge faced by users—ingredient availability. By meticulously matching essential and optional ingredients, the system ensures that recommended tasks align with the user's pantry inventory, empowering them to embark on tasks without the obstacle of missing ingredients.

Finally, the Range Query for Duration Filtering algorithm recognizes the significance of time constraints in task selection. Whether users seek quick recipes for hectic weekdays or leisurely projects for weekends, this algorithm filters tasks based on their duration, offering tailored recommendations that doesn't conflict with the user's schedule and preferences.

In essence, these algorithms work in concert, harmonizing user inputs, task attributes, and system capabilities to deliver a personalized and adaptive task assistance experience. From exploring diverse tasks to accommodating ingredient availability and time constraints, this Conversational Task Agent leverages these algorithms to enrich user interactions and promote continuous progress through every task.

2.1 Search and Indexing

To implement the Conversational Task Agent, it's essential to have efficient search and indexing algorithms and methods. This allows for quick retrieval of information about various manual tasks. The foundation of the system is a robust search index, created to capture all the details of task descriptions, including ingredients and time required.

The core of this search system is the Term-based Search algorithm. It uses BM25 to quickly find tasks that match what users are looking for. The algorithm analyzes task titles, descriptions, and ingredients. This lets users easily explore a wide range of tasks, from cooking to fixing things around the house. It's the first step in providing a great user experience, where recommendations match what users want to do.

Another part of the system is Vector Embedding with KNN. This helps the system understand the meaning behind user searches. It does this by placing tasks in a special space and using K-nearest neighbors (KNN) to find similar tasks. This lets the system not only recommend relevant tasks but also suggest variations based on meaning. This smarter approach helps users find tasks they'll like, considering their past choices.

The system also considers what ingredients users have on hand with Boolean Query for Ingredient Matching. This algorithm makes sure recommendations use ingredients users already have. Whether it's finding a recipe that uses what's in the pantry or suggesting other projects based on available supplies, this feature helps users find tasks they can actually do and makes it easier to complete them.

The system understands that users have limited time, so it includes the Range Query for Duration Filtering algorithm. This lets users filter tasks by how long they take. Whether users want something quick or a longer project, this feature tailors recommendations to their schedule, making it easier to find tasks that fit their day.

All these algorithms and indexing methods work together to make the Conversational Task Agent's search and retrieval system work great. From finding tasks quickly to understanding what users mean and considering what they have on hand, this system helps users explore, discover, and try out different tasks easily. It makes it easier than ever to find the right task and get it done.

3 Evaluation

We carefully examine different search methods used by our Conversational Task Agent to see how well it performs.

For **simple queries with titles and descriptions**[2], the agent easily finds relevant tasks and shows them clearly. For example, if you ask for *"chicken marsala"*, it might recommend tasks like *"How To Make Chicken Marsala at Home"*. This shows the system understands what you're looking for and suggests something that fits.

The agent can also search for keywords in titles and descriptions (**term-based search**[3]). This lets it find tasks that might not be an exact match but are still interesting. For example, searching for *"yogurt"* might give you "How To Grill Juicy, Flavorful Shrimp". This shows the system can be flexible and suggest new ideas.

The system can also consider what ingredients you have on hand. It uses something called *boolean queries* (used for **boolean query search**[4]) to find tasks that use those ingredients. So, if you search for *"chocolate"*, it might recommend something you can make, like "Chocolate Dessert Salami." This helps you find tasks you can actually do with what's already in your kitchen.

Entering the domain of understanding meaning, **embedding description search**[5] takes user interactions to a deeper level of comprehension (demo in Figure 5). It places task descriptions into a continuous space of vectors, going beyond just matching keywords to offer users more tailored recommendations that fit their preferences.For example, if you ask *"how to do chicken marsala"*, it might recommend something specific like "Our step-by-step recipe for classic chicken Marsala." This way, you get recommendations that are more relevant to what you're interested in.

Overall, by testing these different search methods, we see how well the Conversational Task Agent helps users find the manual tasks they're looking for. From finding things quickly to understanding what users mean and considering what they have on hand, each part of the system works together to make it easy to explore and discover new tasks.

3.1 Dataset description

The dataset provided for the Conversational Task Agent project is composed by a collection of nearly 1000 recipes, some of them offering a detailed blueprint for culinary creations and manual tasks. These recipes cover various cuisines, techniques, and ingredients, appealing to different user preferences. From classic dishes to inventive creations, the dataset embodies culinary exploration and do-it-yourself cooking.

Structured in a JSON format, each recipe entry within the dataset has a comprehensive set of attributes, including the recipe name (displayName), a brief description offering insights into the dish's essence,

and images providing visual cues for the final outcome. Furthermore, the dataset dives into the details of ingredient lists, preparation steps, and cooking durations, empowering users to embark on culinary adventures with confidence and clarity.

Moreover, the dataset offers insights into the tools and equipment required for each recipe, ensuring users are well-equipped to execute tasks perfectly. Additionally, attributes such as difficulty levels, servings, and nutritional information giving to the users integrated outlooks of what they are cooking, enabling informed decision-making regarding recipe selection and preparation.

Essentially, the dataset acts as a valuable resource for culinary ideas and practical guidance, aligning with the goals of the Conversational Task Agent project by enabling smooth task retrieval and adaptive assistance. Its structured format and diverse attributes provide a strong basis for developing an intuitive and user-centric task assistance system.

4 Critical discussion

In this part of the report, we want to highlight some of the challenges encountered during the implementation phase and the corresponding solutions devised to address them. One notable obstacle arose when dealing with recipes lacking descriptions, particularly impacting **embedding description search** operations. Initially, we attempted to circumvent this issue by setting empty descriptions (with empty strings). However, this approach proved to be detrimental to system performance, resulting in consistent timeouts. To remedy this, we implemented a strategy of assigning random strings to missing descriptions. By ensuring uniqueness in the description *embeddings*, this adjustment successfully mitigated timeouts, while maintaining the necessary consistency for the results given by the embedding searcher to be coherent and aligned with the user desire.

5 References

1) Illustrated Guide to Transformers Neural Network: A step by step explanation

2) What is a GPT?

3) Hugging Face - NLP Course

6 Attachments

Figure 2: Simple query search and response

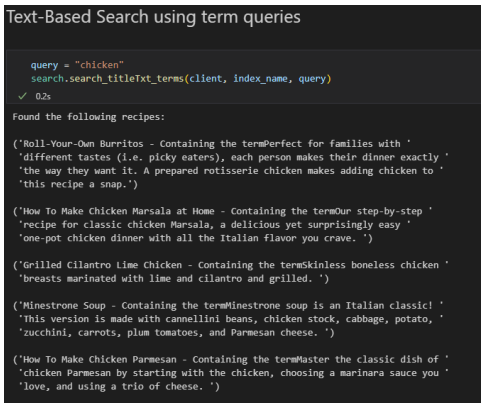

Figure 3: Search using term queries

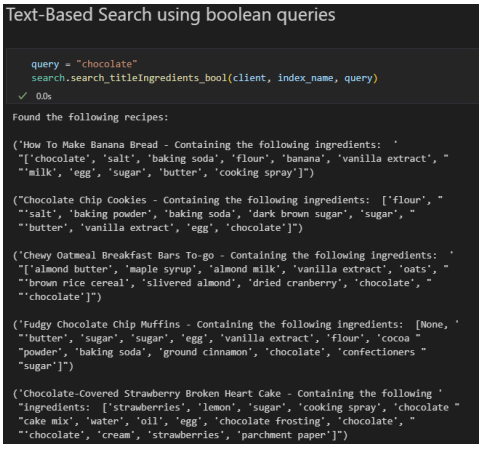

Figure 4: Search using boolean queries

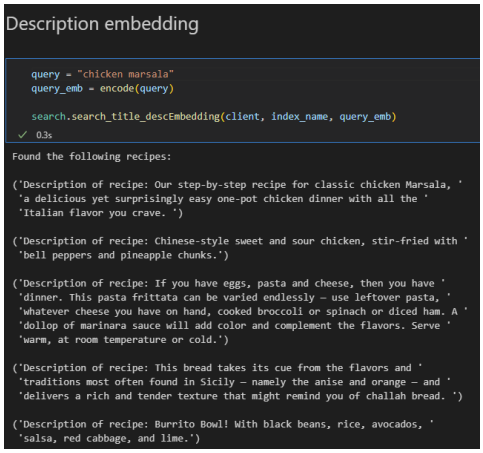

Figure 5: Search using description *embeddings*