Evaluating Diverse Knowledge Sources for Online One-shot Learning of Novel Tasks

James R. Kirk, Robert E. Wray, Peter Lindes, John E. Laird

The Center for Integrated Cognition at IQM Research Institute
24 Frank Lloyd Wright Dr.
Ann Arbor, MI 48105 USA
{james.kirk, robert.wray, peter.lindes, john.laird}@cic.iqmri.org

Abstract

Online autonomous agents are able to draw on a wide variety of potential sources of task knowledge; however current approaches invariably focus on only one or two. Here we investigate the challenges and impact of exploiting diverse knowledge sources to learn, in one-shot, new tasks for a simulated household mobile robot. The resulting agent, developed in the Soar cognitive architecture, uses the following sources of domain and task knowledge: interaction with the environment, task execution and planning knowledge, human natural language instruction, and responses retrieved from a large language model (GPT-3). We explore the distinct contributions of these knowledge sources and evaluate the performance of different combinations in terms of learning correct task knowledge, human workload, and computational costs. The results from combining all sources demonstrate that integration improves one-shot task learning overall in terms of computational costs and human workload.

Introduction

Typical AI systems are designed to exploit one or possibly two sources of knowledge, such as pre-programmed task knowledge (e.g., expert systems), experience in the world (e.g., reinforcement learning), processed human-language artifacts (e.g., large language models/LLMs), general planning capabilities, human-curated domain knowledge (e.g., ontologies), prior examples of task performance (e.g., imitation learning, deep learning), or direct interaction with humans (e.g., interactive task learning). Each knowledge source has strengths and weaknesses. Key dimensions for assessing a knowledge source includes breadth, availability and cost, correctness, and relevance to the agent’s situation.

We hypothesize that an agent that strategically uses multiple, diverse knowledge sources can exploit their strengths while overcoming (or at least minimizing) their weaknesses. We explore this hypothesis via an agent embodied as a mobile robot with a single robotic arm and gripper in a simulated, continuous household environment. Figure 1 illustrates a specific “tidy kitchen” scenario, which is the learning task used throughout the remainder of the paper.

The approach we describe builds on decades of research in cognitive architecture for creating autonomous agents that integrate multiple cognitive capabilities, inspired by the human ability to seamlessly synthesize knowledge from all these sources (and others). Using Soar (Laird 2012), we develop and evaluate a general autonomous agent that exploits three different knowledge sources: domain-general task planning, human instruction, and the GPT-3 LLM (Brown et al. 2020).

We evaluate combinations of these sources on one of the most challenging online learning problems: acquiring the ability to perform wholly novel tasks in one shot. In Figure 1, the agent is learning to tidy a kitchen and the instructor has provided the goal to put away the ketchup. The agent uses this goal instruction, along with planning capabilities, to move the ketchup to the refrigerator. We exploit these capabilities and extend them to include the LLM. The agent uses the LLM to generate possible goals (ketchup in the fridge) and actions (pick-up(ketchup)). Results demonstrate the positive impact of combining these heterogeneous knowledge sources during task learning.

Below we describe previous research on task learning on which this work builds. We provide an analysis of the potential costs and benefits of knowledge sources before analyzing each of the ones used in this project in more detail. We then describe the experiments performed with different combinations of these knowledge sources in online task learning, followed by a presentation and analysis of the results. Finally, we compare and contrast our work with recent attempts to use multiple knowledge sources in task learning.
Learning Novel Tasks

This research builds on research on interactive task learning (ITL), where new tasks are learned online via natural language interaction in one shot (Mininger, Kirk, and Laird 2021, Kirk and Laird 2016). The general scenario we envision is an AI agent acquiring a new task while embedded in an environment, as in Figure 1. The agent has primitive knowledge of its perceptions (a predicate-oriented representation extracted from its environment) and primitive actions (e.g., pick-up, put-down, go-to-location) that involve continuous execution in the environment. Once the agent has learned a new task, it can perform it in similar future situations.

Learning a new task requires acquiring different types of knowledge, determined in part by the type of task being learned. In this paper, we limit our consideration to goal-based tasks where an agent must learn two types of knowledge:

- **Goal knowledge** describes the final state to be achieved. The goal for “tidy the kitchen” is most easily defined as a set of distinct goal conditions of which there are 12 in our agent’s environment, such as “if there is a plastic bottle, it should be in the recycling bin.” For convenience, we will often describe these goal conditions as *subtasks* of “tidy kitchen.”

- **Control knowledge** for selecting actions that achieve the goal (often called a policy).

In the previously-developed ITL agent (Mininger and Laird 2022, Mininger and Laird 2016), the agent interactively acquires additional types of knowledge from the instructor, including knowledge for grounding goal concepts to the agent’s perception of its world (such as what a “clear” block is), for grounding of action concepts to its primitive action knowledge (such as that “deposit” is the same action as “put down” for a block), and decomposing tasks into subtasks. In the longer term, we are also interested in extending this approach to learning these additional types of knowledge as well.

Knowledge Sources

The ITL approach outlined above achieves online, one-shot task learning, but it requires direct, ongoing human guidance during all phases of task learning. We hypothesize that it is possible to supplement the knowledge gleaned from human instruction and planning with other sources. Large language models present an intriguing potential source of knowledge which we explore in this work. However, we also recognize that each of these knowledge sources has different potential benefits and costs that need to be better understood. In this section, we provide an initial analysis of properties of these knowledge sources that might impact their costs and benefits in task learning. These provide hypotheses to be tested in the experiments presented in the second half of the paper, where we evaluate different combinations and their resulting behavior.

Table 1 summarizes the analysis, with the columns listing five properties of knowledge sources, and the two types of knowledge to be learned:

| Knowledge Source | Relevance | Breadth | Correctness | Tractability | Affordability | Goal K | Control K |
|------------------|-----------|---------|-------------|--------------|---------------|--------|-----------|
| Planning         | +         | -       | +           | -            | -             | +      |           |
| Human Instruction| +         | +       | +           | -            | -             | +      | +         |
| LLM              | ?         | +       | +           | -            | +             | +      | +         |

Table 1: Assessment of knowledge sources for task learning

Planning

Once a goal is known, planning can be used to solve it. For learning novel tasks, it is difficult to have task-independent heuristics or cost-functions beyond depth of search. Thus we use iterative deepening so that simple solutions are found quickly and the total planning time is limited by a fixed depth threshold. A side-effect of planning is that control knowledge is learned (via chunking in Soar) that is immediately available for future, similar tasks.

The strength of planning is that it derives task-relevant, actionable control knowledge that is grounded in the agent’s embodiment and environment. It uses internal search via task-independent models of its own actions, which are often inexpensive to obtain. Its breadth is usually limited, as it requires embodiment-specific action models, as well as groundings of concepts and relations in goals to its action models and the world state. The knowledge it produces is
correct (assuming correct action models). Planning is computationally expensive \cite{Bylander1994}, making it less apt as tasks scale in complexity.

The ITL agent also uses means-ends reasoning to execute its actions in novel situations, such as approaching the refrigerator from a new location (as in Figure 1). This reasoning is inherent to the agent’s processing and is necessary for grounding the agent’s interaction with its world no matter what additional knowledge sources are used.

**Human Instruction**

Through natural language, a human can teach the agent a new task by leading the agent through the task \cite{Gluck and Laird2019}. A human can interactively specify goals and subgoals; define novel objects, relations, and properties; and actions to take in specific situations. In the agent described above, the human initially drives the interaction, naming a new task for the agent to perform. The agent then requests a description of the goal (or a procedure for non-goal-oriented tasks), and will request the definition of an unknown word or a known word that the agent cannot ground to the current situation.

The strength of human instruction is that a human often has complete, correct, and contextually-relevant knowledge (grounded in the environment, the agent’s embodiment, and the task). The most significant limitation of human instruction is that it requires a knowledgeable, attentive, and available human, which is not always feasible. Furthermore, relying on a human for every detail of a task can be tedious.

**Large Language Models**

Large Language Models contain unfathomable amounts of knowledge, including knowledge that is potentially relevant to learning new tasks \cite{Ahn et al.2022, Huang et al.2022, Logeswaran et al.2022, Reed et al.2022}. For this paper, we use GPT-3 \cite{Brown et al.2020} for a combination of its breadth, accessibility, and relative affordability\footnote{Total use of GPT-3 for the reported experiments was less than $10 USD.}. We have extended the previously described agent so that it can construct a query (or prompt) to GPT-3 customized for its situation and task-learning goal.

An example prompt using our strategy is shown in Table 2. Everything above the horizontal line is the prompt and everything below it is the response returned by GPT-3. The prompt, including parenthetic delimiters, such as “(EXAMPLES)”, is just text. Even though they are not special commands, they lead GPT-3 to produce structured responses as shown in the example.

We employ a template-based prompting strategy \cite{Wray, Kirk, and Laird2021}. The initial part of the prompt template provides an example task. The example specifies a task to learn (deliver package), the task context (in a mailroom), and an object to execute the task on (package addressed to Gary). This is followed by the response desired from the LLM, which is the goal of the task denoted with (RESULT) and then steps to achieve that goal. The final line of the prompt acts as a template that is instantiated with the task, current location context, and object description:

(TASK) Task name: \texttt{?task}. Task context: \texttt{?current-location}. Aware of \texttt{?object-description}.

The prompt uses language that is consistent with the agent’s language capabilities and its structure “encourages” GPT-3 to provide a relevant response. It is biased toward responses that include actions the agent knows, and responses are processed using the same language parsing and grounding process as used for human instruction. Explorations of different prompting strategies for task learning is the subject of a future paper \cite{Kirk et al. in preparation}.

Using this prompt-creation process, responses from GPT-3 provide both goal and control knowledge. As mentioned earlier, the GPT-3’s breadth offers a significant strength. A goal of this effort is to determine whether it is possible to constrain the context when using the LLM so that the resulting responses are relevant and correct enough to be useful for task learning. In the highly familiar situation of a kitchen, we expect it will produce relevant responses. However, because LLMs generate responses based on statistical patterns of words/tokens in the training set, there is no guarantee that responses will be relevant or correct.

In terms of computational cost, a few queries, parameterized for the agent’s situation, costs little, in either computation or money. Queries that need to be repeated many times, however, can be monetarily costly for complex task learning, and it can be computationally expensive to test and verify responses. Thus, our work seeks approaches that ensure correctness, relevance, tractability, and affordability.

**Combining Knowledge Sources**

As Table 1 suggests, none of the knowledge sources provides all the desired features. Human instruction comes close, but it requires concerted human effort that can be unaffordable. Thus, it is natural to explore how these sources can be combined to mitigate their limitations. Below we discuss how we integrate pairs of sources as well as all three. For each condition, the human provides the name of the task, and then the goal concepts and associated control knowledge must be learned by the agent.

**Planning and Human Instruction**

This combination existed in the original agent that we have extended. The basic process is simple: human instruction is used to acquire descriptions of the goal, and planning is used to search to achieve the goal. When the goal is achieved, control knowledge is learned (via chunking).

If the planning fails (i.e., hits the depth threshold without finding a solution) as in the example below, the agent asks for the human for next step\footnote{All examples in this section are from actual interactions between a human instructor and the agent.}.

**Instructor:** If a plastic bottle is on the counter then the goal is that the plastic bottle is in the bin.

**Agent:** What do I do next for tidy?

**Instructor:** Pick up plastic bottle.
Table 2: Example of an agent-generated prompt and the resulting response from GPT-3 (Temperature=0)

| Agent-created Prompt | GPT-3 Response |
|----------------------|----------------|
| (EXAMPLES)(TASK) Task name: deliver package. Task context: I am in mailroom. Aware of package addressed to Gary, package is in mailroom. (RESULT)The goal is that the package is in Gary’s office (END RESULT) Steps: 1. Pick up package addressed to Gary 2. Go to Gary’s office 3. Put package onto desk in Gary’s office (END TASK)(END EXAMPLES) | (RESULT)The goal is that the plastic bottle is in the recycling bin.(END RESULT) Steps: 1. Pick up plastic bottle from counter 2. Put plastic bottle in recycling bin(ENDER TASK) |

After the agent executes that step, planning is invoked and the process continues until the goal is achieved (bottle in bin). Anecdotally, this combination makes a huge difference when interacting with the robot, as planning is sufficient for solving many tasks and it eliminates much of the drudgery of instructing the robot (e.g., giving it instructions to approach the table to pick up the bottle). However, it still requires instructions to describe the goal and instruct the robot when the planning fails.

Human Instruction and a Large Language Model

In this combination, after the human names the task, the agent constructs a prompt for both a goal and for steps, as was shown in Table 2 for every grabbable object (e.g., the cup) in the scenario, one at a time, providing context of object properties (e.g., paper) and location (e.g., on counter). In the experiments, we used two small prompt examples, “deliver package” (as in the table) and a similar example about storing a package in a closet.

The prompt is generated incrementally starting with the goal followed by steps, generating multiple responses for each. Three responses are created, one temperature 0 response and two temperature 0.9 responses. These are ordered by the average log probability of the tokens in the response. The agent uses the highest probability goal first, but before processing a goal from the LLM, it asks for human confirmation:

Agent: For a glass in the dish rack is the goal that the glass is in the dishwasher and the dishwasher is turned on?
Instructor: No.
Agent: For a glass in the dish rack is the goal is that the glass is in the cupboard and the cupboard is closed?
Instructor: Yes.

If the human responds negatively, the agent retrieves the next response from the LLM and queries the human again. If the human responds in the affirmative, the agent transforms the goal from the LLM into a conditional goal statement for the object (e.g. “If a glass is in the dish rack then the goal is that the glass is in the cupboard and the cupboard is closed.”) and attempts to learn the goal from this natural language description using its existing language and goal learning abilities.

After learning the goal, the agent appends the selected LLM goal response onto the prompt in order to generate steps for that goal. This process continues, where the agent suggests the step the LLM proposes as most probable and the human confirms that response or requests a different response. If the responses from the LLM are ever exhausted, the agent requests the human give their own description:

Agent: Should I “Pick up plate from dish rack?”
Instructor: No.
Agent: What do I do next for tidy?
Instructor: Open cupboard.

In the ideal, the human is only providing confirmation of the goal and task steps. However, if an LLM response is incorrect or irrelevant, the human is there to provide correct and relevant knowledge. The language model has the potential to decrease human effort, but there is no guarantee that its responses will be correct, relevant, or grounded to the specifics of the task and the agent’s embodiment. Thus, the human will need to be attentive, correcting errors and filling in missing steps.

Planning and Large Language Models

In this combination, the agent queries the LLM only for subtask goals and uses planning to search for a solution, using the prompting process described above but only for subtask goals. The most probable response for a goal is used by the agent. If the goal retrieved from the LLM is correct, the agent finds a solution. If the goal is not correct but is achievable (in the current situation), the agent finds a solution, but the correct goal for the subtask is not achieved. If the goal retrieved is not correct and is unachievable (e.g. putting an object in a drawer, where no drawer exists), then the agent is unable to recover.

Thus, in some cases, we predict this combination will be effective without any human instruction (other than naming the task), but whenever the LLM falls short, the result will be failure. However, it is difficult to predict a priori how such often failures will occur and we explore this question in the subsequent experiments.
Combined

When all knowledge sources are available, the approach is similar to the Instruction and LLM combination, but the agent uses planning to avoid asking for steps when possible.

The combination of all methods should result in more correct outcomes than the planning and LLM condition and require substantially less human intervention than the other combinations that require human instruction. Assuming this general analysis is correct, the most interesting question is the extent to which human instruction can be replaced by the LLM when it is combined with planning and the human. If it substantially reduces the amount of required human instruction and the human effort required, this result would be a compelling case for incorporating LLMs in task learning (and potentially other cognitive agent learning problems).

Assessing Combinations of Knowledge Sources

The combination of knowledge sources in the agent, as described above, affords the ability to explore the properties of various combinations of these sources. In this section, we describe the design and results of an experiment that evaluates these sources for online, one-shot learning of a task.

Experiment Design

The experiment is situated within a household environment and a mobile robot built on top of the Magicbot simulator. As outlined above, the mobile robot has a single arm and manipulator. In the experiment, the robot can grasp and manipulate all the objects relevant to the task to be learned (i.e., the task is achievable with its primitive level capabilities). The agent, as described above, extends a previously-developed interactive-task-learning agent implemented in Soar (Kirk 2019; Kirk and Laird 2019; Mininger 2021).

Learning Task

The task to be learned in the experiment is to tidy a kitchen. In all of the experimental conditions, the agent initiates an interaction by asking what task it should learn and the human instructor responds with “tidy kitchen.” Figure 2 illustrates the scenario used in the experiment. Twelve objects are located in various places in the room, such as a napkin on the table, a plate in the dish rack, etc.

The goal of the task is that each of twelve objects must be in an appropriate location and the agent must learn goals for each object. The experiment is designed so that the goal location for each type of object is determined by the type of object, its location, and its properties (e.g., the mug on the table goes into the dishwasher, the glass in the dish rack should be put in the cupboard and not the dishwasher). The agent can determine object types and properties from its perceptual representation, but additional external knowledge is needed to determine the goal location for each type.

Executing some actions requires the one-armed robot to perform extra steps because it cannot open doors while holding an item. This situation is suggested in Figure 1, where the first step in the robot’s plan it to open the door to the refrigerator. While the agent’s planning knowledge handles this limitation, we expect that responses from an LLM will sometimes omit steps (i.e., in vernacular language, “put the ketchup in the fridge” implies opening the door). Further, even if the prompting strategy can make these steps explicit, the LLM’s apparent assumption of an embodiment and resultant affordances may more closely match those of human actors than those of a mobile robot (Ahn et al. 2022).

In addition to the relocation of objects, three of the locations in the environment have doors that must be closed for successful task completion. For instance, the agent must learn to put the ketchup in the fridge and to close its door after opening it. In the “tidy kitchen” task, achieving the correct position for 6 objects require 2 assertions (e.g., “milk in the fridge” and “fridge door is closed”) for a total of 15 goal assertions to be achieved (some of the assertions overlap). In all, the one-shot learning task is to learn an executable representation of the tidy-kitchen task that allows the agent to perform the task in the same (or similar) environment in the future without additional instruction or planning.

Experimental Conditions

The experimental conditions are based on combinations of knowledge sources outlined in the previous section. For all experiments, the specific GPT-3 model used is text-davinci-002 and TopP is set to 1. Temperature varies between 0 and 0.9 as previously described. As mentioned earlier, all these conditions include means-end reasoning for executing predefined low-level tasks, such as “approach.”

- **Planning and Human Instruction**: The agent receives goal descriptions from the human, but otherwise uses its general planning knowledge to search for a solution. For this condition, the search depth is limited to 4. This is sufficient for all the individual tasks (i.e., 100% task completion rate), thus there is no human instruction on actions. In the results section, this condition is called “Planning” because human instruction is limited to goal statements.

- **Human Instruction**: The human provides the agent goal descriptions (“milk in fridge”) as well as action descriptions (“pick up milk”).

- **Planning + LLM**: Other than the initial “tidy kitchen” provided by the human, in this condition the agent creates prompts to elicit goal descriptions from the LLM and seeks to carry them out using the planning capability without further human input. This condition uses the
LLM responses with the highest probability. We expect that this condition will result in some failures when there is a mismatch between LLM responses and the agent’s embodiment and existing knowledge.

- **Human Instruction + LLM** The agent attempts to query the LLM for goals and actions and uses human input to confirm/disconfirm responses or provide descriptions if all suggestions are exhausted. Similar to the Human Instruction condition above, planning is used for only low-level planning of approaches/movements.

- **Planning (Depth 2) + Inst. + LLM** In this condition, all three sources of knowledge are used to support task learning. The LLM is prompted for both goals and actions. Human input is used to confirm/disconfirm LLM suggestions and provide descriptions if all suggestions are exhausted. Planning is used to attempt to find the goal state. Planning depth is limited to 2 steps.

- **Planning (Depth 4) + Inst. + LLM**. Same as the previous condition except that the planning depth is limited to 4.

**Measures** To evaluate the alternatives, we look to the properties introduced in Table 1 and suggest measures for each, below (We did not attempt to measure breadth). Our expectation is that the outcome of the experiment will be consistent with the analytical assessment made earlier and also reduce the uncertainty in some of the properties of LLMs as applied to one-shot task learning.

- **Relevance**: For these experiments, the planning and human instruction by their nature are always relevant. We measure relevance in the LLM conditions as the percentage of confirming (“yes”) responses within the total number of yes/no confirmation queries to the human.

- **Correctness**: Task completion rate is used as a proxy for correctness. Task completion rate is computed as the number of correct assertions / total number of goal assertions (15, as above).

- **Tractability**: Planning is the primary element in the system that could cause tractability issues. We report the total number of search states explored by planning to capture tractability. By design, this experiment only requires shallow search to find plans and we do not expect tractability to be an issue.

- **Affordability**: Human time/resources are the largest factor impacting affordability in these experiments. We assess affordability by a number of different measures:
  - Instructions: Total number of instructions provided by the user.
  - Words: Some instructions are more complex than others, so we also report the total number of words.
  - Yes/No Responses: With the instructor+LLM conditions, the instructor’s role shifts from generating goals and actions to assessing the responses of the LLM. We thus report the number of these yes/no responses.

**Results**

Overall results for the “tidy kitchen” task (as in Figure 2) are illustrated in Table 3 and replicated in graph form in Figure 3. Below, we discuss individual results for each measure.

| Condition | Completion Rate (%) | Rel. Responses (%) | Search States | Instructions | Words | Yes/No Instructions |
|-----------|---------------------|--------------------|---------------|--------------|-------|---------------------|
| Planning | 100.0               | N/A                | 404           | 14           | 262   | N/A                 |
| Instruction | 100.0               | N/A                | 55            | 50           | 380   | 0                   |
| Plan+LLM | 46.7                | 57.1               | 347           | 1            | 2     | N/A                 |
| Inst+LLM | 100.0               | 66.1               | 35            | 71           | 205   | 62                  |
| Combined | 100.0               | 50.0               | 345           | 42           | 178   | 32                  |
| Combined | 100.0               | 30.4               | 230           | 30           | 125   | 23                  |

Table 3: Summary of experimental results.

**Completion Rate** In all conditions except the Plan+LLM condition, the agent fully satisfied all of the goals. Further, for these fully-completed conditions, the agent, on being presented a similar scenario, completes “tidy kitchen” without additional planning, instruction, or use of the language model. Thus, the tidy kitchen tasks are learned in a single shot as outlined previously.

In the Plan+LLM condition, the agent fully completes only four of the goal concepts (one is partially satisfied, resulting in the correct location but an open door). The prompting strategy is mostly insensitive to increases in temperature for GPT-3, resulting in similar responses from the LLM even as temperature (randomness) is increased.

**Relevant Responses** The instructor accepted between a half to about a third of the responses generated by the LLM. This outcome overall is somewhat poorer than recent work focused on task learning from LLMs (see Related Work), but is achieved without any fine-tuning for this domain.

The relevancy of the combined depth=4 condition is much lower (30%) than the relevancy of the other LLM conditions. This lower value results from the combination of the human feedback and the increased planning depth. Goals from the LLM are rejected until a satisfactory goal is provided. At that point, the planning capability can produce a complete set of actions (as it did for the first experimental condition). Thus, lower relevancy in this condition results from the use of the other knowledge sources, which reduce the need to further query the LLM for additional (and more often acceptable on the first try) action-focused responses. The relevancy of the Plan+LLM condition would be lower if it had not failed halfway through the task.

**Affordability Measures** With the addition of the LLM, instructions increase by 40% (Inst.+LLM) or decrease by 15-40% (Combined cases). Because the LLM’s role is to supplement instructor input, this somewhat moderate decrease may at first seem inconsistent with our expectations. However, the total number of words produced by the instructor decreases substantially in all cases, by 46% (Inst.+LLM), 53% (Combinedd_{st}), and 67% (Combinedd_{st}). These reductions largely come about because the agent via the LLM pro-
produces candidate goals and actions, which the instructor can then accept or reject via simple yes/no instructions. The last column of the table provides totals for yes/no responses. In the “Instruction” condition, there are no yes/no instructions. The other LLM conditions range from 23-62.

Figure 4 summarizes the shift from more complex instructions in non-LLM conditions to less complex instructions in the LLM ones. In the non-LLM conditions, the human instructor provides goals (and actions if required) for each subtask. In the LLM conditions, the responsibility of the instruction shifts significantly toward confirming or disconfirming suggestions from the LLM. In other words, the LLM is requiring that instructor produce many fewer goal and action instructions in these conditions. Compared to the “Instruction” condition, goal instructions (the most complex instructions) are reduced in the Combined condition by 42% (7 vs. 12) and by 58% in Combinedd_4.

Overall, these experimental results are consistent with the analysis and experimental hypothesis summarized in Table 1. The combination of planning, instruction, and the LLM led to task learning that required less human interaction and no increase in the computational costs of planning. The responses of the LLM taken in isolation were, in terms of correctness and relevancy, unexceptional, possibly marginal. While we will explore further improving response generation, the more important outcome from our perspective is that even marginal responses from the LLM, when coupled with simple yes/no feedback from the human instructor, enabled 100% task completion while also substantially reducing human interaction requirements.

Related Work

In just a few years, research on LLMs and knowledge extraction proceeded from extraction of simple relations, disconnected from agent use (Bosselut et al. 2019), to illustrations of robotic task learning in simulated and physical environments (Ahn et al. 2022; Huang et al. 2022; Logeswaran et al. 2022; Reed et al. 2022). We focus on these latter examples in Related Work. Generally, the research reported in these papers is more focused on extracting task knowledge from LLMs exclusively in comparison to our emphasis on 1) integrating multiple knowledge sources and 2) learning both control/policy knowledge and goal descriptions.

Ahn et al. explore extracting knowledge from LLMs for physically-grounded tasks in an embodied robot. Their SayCan system uses context from the situation and robot affordances to bias the LLM toward relevant responses (using “relevance” as outlined previously). Focusing prompts on the current situation and known aspects of the embodiment is similar to our LLM prompting approach. The responses from the LLM trigger action planning. SayCan uses RL to learn mappings between robotic affordances and actions, but not new goals. This is in contrast to our approach, where the affordances and robotic primitives are fixed but new tasks/goals are learned.

Logeswaran et al. investigate using language models to predict plans for real-world environments, using context from the environment to rank LLM predictions and improve the relevance of retrievals. Unlike our work, they perform few-shot learning, not one-shot, and require training data for learning subgoals and the ranking model rather than learning entirely online.

Huang et al. explore how LLMs can be used for planning in embodied robotic environments. They investigate injecting feedback from the environment and a human into LLM prompts in order to improve instruction completion. Like us, they focus on integrating human-LLM-planning knowledge for task learning. Unlike our work, they emphasize the dynamic nature of the interaction, including allowing the human to interrupt LLM-based planning. However, this dynamic interaction is the goal of their work. Task learning is not one shot (performance on “unseen” tasks ranges from very low to 76%). They also do not evaluate the contributions of different knowledge sources or their combinations.
Discussion and Conclusions

LLMs are a promising tool for task knowledge acquisition. However, in today’s usage (including the related work), they are not yet able to be used reliably by themselves for task learning. We explored the use of additional sources of knowledge to provide context, to guide LLM generation, to constrain and to verify those responses. Use of additional knowledge sources is critical to enabling learning task knowledge that is relevant, correct, and actionable for a specific situation that a robot or other agent may encounter. Furthermore by using LLMs, task-learning agents can reduce costs associated with learning, including human instruction and computation. There are also trade-offs from using different combinations of knowledge sources. We demonstrated an agent that learns a task in one shot from many sources in combination (human instruction, LLM, planning). Having learned the task in one shot, the agent performs the task in the future without any further instruction or planning.

For future work, we plan to explore more diverse domains and complex tasks that involve more interactions between objects, rather than tasks where each object can be handled independently. This will lead to a computational explosion in planning costs, and will further the need to rely more heavily on other knowledge sources like the LLM and human. Taking inspiration from [Huang et al.] we plan to explore injecting more interaction and context of failures into prompts in order to illicit better LLM responses when the agent retrieves knowledge that is not correct or relevant. We also plan to explore utilizing other sources of knowledge, including knowledge bases such as WordNet [Fellbaum 1998] and ConceptNet [Speer, Chin, and Havasi 2017]. We have performed initial investigation into using WordNet to identify synonyms of words, so that more responses from the LLM can be understood by the agent.

Acknowledgments

This work was supported by the Office of Naval Research, contract N00014-21-1-2369. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Department of Defense or Office of Naval Research. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon. We would also like to thank contributors to the development of the ITL agent: Mazin Assanie, Elizabeth Goeddel, Aaron Mininger, Shiwil Mohan, and Preeti Ramaraj.

References

Ahn, M.; Brohan, A.; Brown, N.; Chebotar, Y.; Cortes, O.; David, B.; Finn, C.; Gopalakrishnan, K.; Hausman, K.; Herzog, A.; Ho, D.; Hsu, J.; Ibarz, J.; Ichter, B.; Irpan, A.; Jang, E.; Ruan, R. J.; Jeffrey, K.; Jesmondt, S.; Joshi, N. J.; Julian, R.; Kalashnikov, D.; Kuang, Y.; Lee, K.-H.; Levine, S.; Lu, Y.; Lui, L.; Parada, C.; Pastor, P.; Quiambao, J.; Rao, J.; Rettinghouse, J.; Reyes, D.; Sermanet, P.; Sievers, N.; Tan, C.; Toshev, A.; Vanhoucke, V.; Xia, F.; Xiao, T.; Xu, P.; Xu, S.; and Yan, M. 2022. Do As I Can, Not As I Say: Grounding Language in Robotic Affordances. ArXiv:2204.01691 [cs].

Bosselut, A.; Rashkin, H.; Sap, M.; Malaviya, C.; Celikyilmaz, A.; and Choi, Y. 2019. COMET: Commonsense Transformers for Automatic Knowledge Graph Construction. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. ArXiv: 1906.05317.

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; Agarwal, S.; Herbert-Voss, A.; Krueger, G.; Henighan, T.; Child, R.; Ramesh, A.; Ziegler, D. M.; Wu, J.; Winter, C.; Hesse, C.; Chen, M.; Sigler, E.; Litwin, M.; Gray, S.; Chess, B.; Clark, J.; Berner, C.; McCandlish, S.; Radford, A.; Sutskever, I.; and Amodei, D. 2020. Language models are few-shot learners. arXiv preprint, arXiv:2005.14165.

Bylander, T. 1994. The computational complexity of propositional STRIPS planning. Artificial Intelligence, 69(1): 165–204.

Fellbaum, C., ed. 1998. WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press.

Gluck, K.; and Laird, J., eds. 2019. Interactive Task Learning: Agents, Robots, and Humans Acquiring New Tasks through Natural Interactions, volume 26 of Stringmann Forum Reports. Cambridge, MA: MIT Press.

Huang, W.; Xia, F.; Xiao, T.; Chan, H.; Liang, J.; Florence, P.; Zeng, A.; Tompson, J.; Mordatch, I.; Chebotar, Y.; Sermanet, P.; Brown, N.; Jackson, T.; Loo, L.; Levine, S.; Hausman, K.; and Ichter, B. 2022. Inner Monologue: Embodied Reasoning through Planning with Language Models. ArXiv:2207.05608 [cs].

Kirk, J.; Wray, R.; Laird, J.; and Lindes, P. in preparation. Improving Language Model Prompting in Support of Semi-autonomous Task Learning.

Kirk, J. R. 2019. Learning Hierarchical Compositional Task Definitions through Online Situated Interactive Language Instruction. Ph.D. Thesis, University of Michigan, Ann Arbor.

Kirk, J. R.; and Laird, J. E. 2016. Learning General and Efficient Representations of Novel Games Through Interactive Instruction. ISBN 0021-9967.

Kirk, J. R.; and Laird, J. E. 2019. Learning Hierarchical Symbolic Representations to Support Interactive Task Learning and Knowledge Transfer. 6095–6102.

Laird, J. E. 2012. The Soar Cognitive Architecture. Cambridge, MA: MIT Press.

Logeswaran, L.; Fu, Y.; Lee, M.; and Lee, H. 2022. Few-shot Subgoal Planning with Language Models. In Proceedings of the NAACL 2022. arXiv. ArXiv:2205.14288 [cs].

Mininger, A. 2021. Expanding Task Diversity in Explanation-Based Interactive Task Learning. Ph.D. Thesis, University of Michigan, Ann Arbor.

Mininger, A.; and Laird, J. 2022. A Demonstration of Compositional, Hierarchical Interactive Task Learning.

Reed, S.; Zolna, K.; Parisotto, E.; Colmenarejo, S. G.; Novikov, A.; Barth-Maron, G.; Gimenez, M.; Sulsky, Y;
Kay, J.; Springenberg, J. T.; Eccles, T.; Bruce, J.; Razavi, A.; Edwards, A.; Heess, N.; Chen, Y.; Hadsell, R.; Vinyals, O.; Bordbar, M.; and de Freitas, N. 2022. A Generalist Agent. ArXiv:2205.06175 [cs].
Speer, R.; Chin, J.; and Havasi, C. 2017. ConceptNet 5.5: an open multilingual graph of general knowledge. In Proc. of the 31st AAAI Conference on Artificial Intelligence, AAAI’17, 4444–4451. San Francisco, California, USA: AAAI Press.
Wray, R. E.; Kirk, J. R.; and Laird, J. E. 2021. Language Models as a Knowledge Source for Cognitive Agents. In Proceedings of the Ninth Annual Conference on Advances in Cognitive Systems. Virtual.