LMRL-GYM: BENCHMARKS FOR MULTI-TURN REINFORCEMENT LEARNING WITH LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Large language models (LLMs) provide excellent text-generation capabilities, but standard prompting and generation methods generally do not lead to intentional or goal-directed agents and might necessitate considerable prompt tuning. Even the best current LLMs rarely ask clarifying questions, engage in explicit information gathering, or take actions that lead to better decisions after multiple turns. Reinforcement learning has the potential to leverage the powerful modeling capabilities of LLMs, as well as their internal representation of textual interactions, to create capable goal-directed language agents. This can enable intentional and temporally extended interactions, such as with humans, the emergence of complex skills such as persuasion, and long-horizon strategic behavior, such as in the context of games. Enabling this requires the community to develop reliable reinforcement learning algorithms for training LLMs. Developing such algorithms requires tasks that can gauge progress on algorithm design, provide accessible and reproducible evaluations for multi-turn interactions, and cover a range of task properties and challenges in improving reinforcement learning algorithms. Our paper introduces the LMRL-Gym benchmark for evaluating multi-turn RL for LLMs, together with an open-source research framework for getting started on multi-turn RL with offline value-based and online policy-based RL methods. Our benchmark consists of 3 Interactive Dialogue tasks and 5 RL Capability tests for a total of 8 tasks, which require multiple rounds of language interaction and cover tasks in open-ended dialogue and text games.

1 INTRODUCTION

Large language models (LLMs) have demonstrated remarkable abilities when naturally conversing with humans (OpenAI, 2023; 2022; Touvron et al., 2023; Google, 2023), answering questions and responding to requests (Shuster et al., 2022b;a; Qin et al., 2023), and even performing coding tasks (Chen et al., 2021b; Wang et al., 2023b). Many of these capabilities are enabled by learning to emulate humans from large datasets of text from the web (Völske et al., 2017; Shuster et al., 2022a; Yao et al., 2023), learning from examples "in context" (Brown et al., 2020), as well as learning from other sources of supervision such as instruction datasets (Mishra et al., 2022; Wei et al., 2022; Wang et al., 2022b) and preference fine-tuning with RLHF (Ziegler et al., 2020; Ouyang et al., 2022). However, directly applying LLMs in settings that require planning or multi-turn interactions presents new challenges. LLMs are not explicitly goal-directed, as they are not optimized to directly solve particular tasks, but rather to produce text that resembles the distribution of human-provided examples or accords with human preferences (Ziegler et al., 2020; Stiennon et al., 2020; Wu et al., 2021; Bai et al., 2022a). This challenge is apparent in solving temporally extended tasks, such as multi-turn dialogue (Irvine et al., 2023; , FAIR), complex tool use (Wang et al., 2022a), multi-step games (Hendrycks et al., 2021b), and other interactive applications. In principle, LLMs should contain the knowledge necessary to succeed in such settings: if the multi-turn interactions center around problem domains that are well represented in the model's training data (such as dialogue), well-trained LLMs should already serve as powerful predictive models in such settings. However, leveraging this predictive knowledge to derive effective actions and strategies requires not just emulating humans, but also planning and optimization.

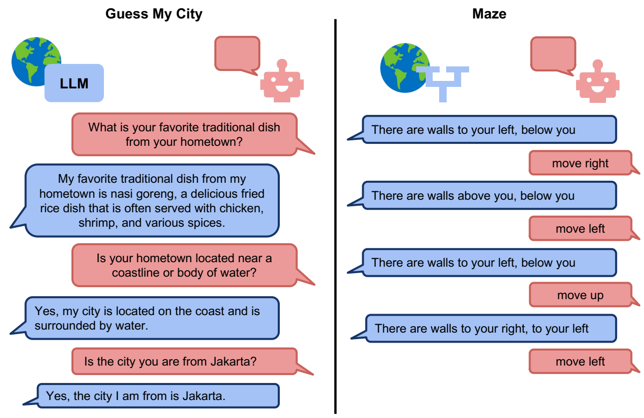

Figure 1: **Overview of LMRL-Gym:** We show sample trajectories from two tasks in our benchmark. In the Guess My City task, the agent learns to ask questions to guess the city the oracle is from, while in the Maze task, the agent learns to make the correct moves based on cues from the oracle.

Multi-turn reinforcement learning (RL) (Sutton & Barto, 2018) in principle offers a path to enable LLMs to do just that. RL could enable goal-directed reasoning and planning in interactive multi-turn settings, including complex dialogue, games, and tool use. We hypothesize that RL could serve as a powerful tool for LLM training, not only for training models to accord with human preferences, but more generally to accomplish tasks in an intentional and goal-directed manner. Text generation can be viewed as a sequential decision-making process, treating a sequence of tokens as a trajectory. Many tasks, such as successfully answering questions or eliciting a desired reaction from a user, can then be framed as optimizing some reward function over these trajectories. However, despite extensive interest in RL for LLMs in recent years, much (though not all) of the recent research in this area has focused on "single-step" RL problems, where a single response is optimized for some quality metric, typically derived from human preference signals (Stiennon et al., 2020; Ziegler et al., 2020; Ouyang et al., 2022; Bai et al., 2022a; Anthropic, 2023; Ramamurthy et al., 2023; Christiano et al., 2023; Casper et al., 2023).

While some works have sought to apply RL for multi-turn tasks (Singh et al., 1999; Li et al., 2016; Shah et al., 2016; Kwan et al., 2022), particularly for goal-directed dialogue (Lewis et al., 2017; Verma et al., 2022), there has been comparatively little research on improving the underlying RL algorithms and very little head-to-head comparison on same sets of tasks. This is perhaps unsurprising: it is easier to evaluate improvements to algorithms for single-turn text generation as compared to multi-turn generation. Multi-turn dialogue requires an interactive evaluation procedure rather than just a static dataset. There is no established protocol for such evaluations, and the "gold standard" constitutes costly and time-consuming studies with human participants.

In this work, we aim to address this challenge and make it possible for RL algorithm researchers to iterate on developing better RL methods for multi-turn language-based interaction tasks, such as dialogue and games. We posit that benchmarking RL algorithms for LLMs presents a very different set of challenges and merits a different set of solutions compared to other benchmarks in NLP. While most NLP benchmarks are based on standard supervised machine learning paradigms, with a training set and a test set (Marcus et al., 1993; Tjong Kim Sang & De Meulder, 2003; Socher et al., 2013; Rajpurkar et al., 2016; Wang et al., 2019; Williams et al., 2018), RL benchmarks require simulators that the trained agents can interact with to measure their performance. In this paper, we use an LLM to simulate a conversation partner in dialogue tasks. While the behavior of the LLM may deviate from human behavior, we verify in a human study in Appendix A that our LLM simulators produce natural text reflecting human norms of conversation. However, our goal is *not* to utilize this approach to benchmark whether LLMs are *good at talking to humans*, but rather as a way to test RL algorithms with datasets that are sufficiently difficult and complex to gauge how effective they might be if they were then trained on data from real humans. Specifically, our benchmark aims to rigorously stress-test the ability of RL algorithms to enable complex goal-directed behaviors in LLMs. To this end, LMRL-Gym also includes a set of text-based strategy games, in addition to the dialogue tasks, that are aimed at providing a more controlled and focused diagnostic assessment of specific RL capabilities.

Our proposed benchmark, LMRL-Gym, consists of 8 tasks. Three tasks are Interactive Dialogue tasks designed to simulate real-world interactions with humans, requiring information gathering (20 Questions, Guess My City) and negotiation (Car Dealer). Five tasks are RL Capability Tests, which are text games designed to isolate specific capabilities of RL training. Each task comes with an offline dataset that can be used for offline RL training, and a "simulator" that can be used to evaluate the performance of the agents in multi-turn interactive tasks. We provide a research framework and toolkit for researchers and practitioners to get started with multi-turn RL for LLMs. This framework includes implementations of PPO (Schulman et al., 2017), ILQL (Snell et al., 2022a), and several baseline methods, implemented in an extensible way designed for future development of tasks, experimentation, and algorithm design.

2 Related Works

Datasets, benchmarks, and libraries. Benchmarks and datasets have been an important factor for driving progress in NLP in domains that include machine translation (Tiedemann, 2012; Bojar et al., 2016), natural language understanding (Rajpurkar et al., 2016; Wang et al., 2019; Hendrycks et al., 2020; 2021a; Ramamurthy et al., 2023), and solving math problems (Cobbe et al., 2021). However, these tasks generally do not involve multi-turn interaction and do not come with rewards, making them hard to adapt to RL research. For example, the standard for evaluating dialogue agents has been to run a human subjects study, but this is time-consuming and costly. Some works have proposed text games for evaluating language-based agents (Chevalier-Boisvert et al., 2018; Hausknecht et al., 2019; Yuan et al., 2019; Fan et al., 2020; Hausknecht et al., 2020; Guo et al., 2020; Ammanabrolu et al., 2020; Yao et al., 2020; Hendrycks et al., 2021b; Singh et al., 2021; Wang et al., 2022a; Yao et al., 2022; Jansen & Côté, 2022; Yao et al., 2023; Zhang et al., 2023; Gontier et al., 2023) and interactive dialogue (De Bruyn et al., 2022b;a). Our aim is to cover a variety of problem settings that reflect challenges in open-vocabulary interaction in addition to text games, that also specifically evaluate offline RL capabilities, which is not done by prior works. Motivated by successes in using LLMs to generate synthetic data (Hausknecht et al., 2019; Park et al., 2023; Bai et al., 2022b), our proposed tasks are based on synthetic data. While such data may differ from natural text, the scope of our benchmark is specific to evaluating RL algorithms, not the ability to interact with humans.

RL for language models. RL for language models has seen success in aligning LLMs with human preferences (RLHF) (Ziegler et al., 2020; Stiennon et al., 2020; Bai et al., 2022a;b; Ouyang et al., 2022; Christiano et al., 2023), optimizing non-differentiable objectives for machine translation (Wu et al., 2016; Nguyen et al., 2017; Kiegeland & Kreutzer, 2021), generation (Tambwekar et al., 2019; Pang & He, 2021; Pyatkin et al., 2022), dialogue (Cuayáhuitl et al., 2015; Georgila & Traum, 2011; Li et al., 2016), question answering (Pyatkin et al., 2022), and summarization (Paulus et al., 2017; Böhm et al., 2019; Wu & Hu, 2018). These include RL methods that learn by directly interacting with the environment (online RL) (Carta et al., 2023) and RL methods that only use a static dataset (offline RL) (Jaques et al., 2020; Snell et al., 2022a; Jang et al., 2022; Verma et al., 2022; , FAIR). However, many of these works operate in the singe-step bandit setting, and do not consider multi-turn goal-directed tasks. Our benchmark, on the other hand, focuses on tasks involving multiple turns of interaction with clearly defined goal-based reward functions.

Capabilities of LLMs. There has been a surge in the capabilities of LLMs for generation (Ghazvininejad et al., 2017; Radford et al., 2019), dialogue (Lewis et al., 2017; Jaques et al., 2017; Shuster et al., 2022b; Snell et al., 2022b), question answering (Pyatkin et al., 2022), summarization (Paulus et al., 2017; Böhm et al., 2019; Wu & Hu, 2018), text-based games (Narasimhan et al., 2015; Hausknecht et al., 2019), translation (Gu et al., 2017), and more. However, these are often supervised learning tasks that do not test the LLMs' abilities to achieve a specific long-term objective. Research on dialogue generation (Jaques et al., 2017; He et al., 2018; Shuster et al., 2022b;a) has often focused on generating feasible-looking agent dialogue without explicit consideration for some multi-turn objective. Our benchmarks allow for the development of algorithms that enable LLMs to *interact* with an environment to achieve long-term objectives, by providing tasks with online simulators and offline datasets.

3 Multi-Turn Generation with RL and Language Models

This section introduces the conceptual foundations of using reinforcement learning for multi-turn generation with language models. We introduce a definition of the Markov decision process for language and a framework for the methods we focus on in this paper.

Definitions. We formalize language generation tasks as a partially observable Markov decision process. We define the state to be the history of tokens and an action as the next token generated by the model. An observation is a single token o_i in the history. The probability of generating the next token is dependent on all of the previous observation tokens o_i. Therefore the Markovian state s is formed by the concatenation of all the previous tokens $[o_0, \ldots, o_i]$. A policy π defines the agent's behavior by taking in the current state s and outputting a new action token a to get s_{i+1}. The environment assigns a reward $r(s, a)$ based on the entire sequence of tokens so far. The tokens in the state are either generated by the policy π or the environment. For example, in the Car Dealer task, the policy generates the tokens for the Seller's utterance and the environment generates the tokens for the Buyer. The full history of their conversation would form the state. A complete sequence of tokens is referred to as a trajectory $\tau = o_0, \ldots, o_T$. The goal of RL is to produce a policy π^* that maximizes the expected discounted sum of rewards over trajectories (τ) under the policy $\pi^* = \arg\max_\pi \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T-1} \gamma^t r_t(s_t, a_t)\right]$, where τ represents the trajectory.

RL Algorithms. Several possible RL algorithms could be used to train language models for multi-turn tasks (Jaques et al., 2020; Verma et al., 2022; Snell et al., 2022a; Schulman et al., 2017; Stiennon et al., 2022; Bai et al., 2022a; Casper et al., 2023). Policy gradient methods, such as PPO (Schulman et al., 2017), directly compute the gradient of the RL objective with respect to the model parameters. Value-based methods estimate a state-action (Q) and/or state-value (V) function. The state-action or state-value function forms a policy by either 1) acting greedily with respect to the Q-function or 2) perturbing the base model's logits with the learned action-value functions (Snell et al., 2022a). RL methods for training LLMs can be *online* or *offline*. Online methods repeatedly interact with the environment, collecting additional data during training. Offline RL instead learns to extract the best behaviors from an existing, potentially suboptimal dataset. Due to the large amount of existing text interactions on the internet, offline RL is an ideal setting for training language models. Therefore, our work primarily focuses on benchmarking offline RL algorithms. However, our tasks also fully support online RL and we include an online PPO baseline in our evaluation.

4 The LMRL-Gym: Synthetic Benchmarks for RL with Language

Our benchmark consists of 8 tasks grouped into two categories: RL Capability tasks and Interactive Dialogue tasks. The RL Capability tasks focus on desirable capabilities for RL algorithms for LLMs such as strategic decision-making, credit assignment, trajectory stitching, partial observability, and use of complex language. For the interactive dialogue tasks, we model them after real-world interactions with humans, such as persuading someone to buy a car or playing a guessing game.

Below, we define the Interactive Dialogue tasks, describe the specific capabilities of RL algorithms for LLMs that our benchmark aims to evaluate through RL Capability tasks, and summarize the data generation and simulation process. We have provided example trials for each task are shown in Figure 4, and a concise summary of the dataset and task statistics in Table 1. The number of trajectories and the average length of the trajectories varies based on the complexity of the tasks.

4.1 Interactive Dialogue Tasks

The Interactive Dialogue Tasks aim to simulate real-world goal-oriented dialogues. We focus on tasks where the agent must make inferences about persuasive strategies and actively gather information by asking questions. Instead of generating these interactions with humans, we generate such interactions through simulating LLMs inspired by successes in using LLMs to generate synthetic data. While the LLM might not be as realistic as a real human, we have found that human raters evaluated the LLM-generated text as quite realistic in most cases, as discussed in our user study in Appendix A. You can find examples from the trained models in Appendix I.

20Qs (Twenty Questions). This task tests whether an agent can gather information about an unknown subject through twenty yes or no questions. The agent must use semantic knowledge of the object to infer the correct answer.

Guess (Guess My City). The Guess My City task performs more complex forms of information gathering, involving open-ended questions about a city. This task evaluates semantic knowledge of a specific city and the agent's ability to parse information from a free-form answer.

Car Dealer. The Car Dealer task tests the ability of RL algorithms to learn successful car sale strategies. This involves decision-making and credit assignment as different persuasion strategies must be adopted for different kinds of buyers.

4.2 RL CAPABILITY TASKS

A central objective of our benchmark is to evaluate the core capabilities that RL enables in large language models. The RL Capability tasks are text-based games designed to isolate specific RL capabilities and are language analogs of tasks where RL is known to succeed. These tasks include Chess, Endgames, Wordle, Maze, and Text-Nav. Below we explain the tasks and the motivation for including them as tests for RL capabilities. Further details on task design for RL Capability tasks can be found in Appendix B.

	Strategic Decision Making	Complex Language	Credit Assignment	Partial Observability	Trajectory Stitching
Maze FO	✗	✗	✓	✗	✓
Maze PO	✗	✗	✓	✓	✓
Text-Nav FO	✗	✓	✓	✗	✓
Text-Nav PO	✗	✓	✓	✓	✓
Wordle	✓	✗	✗	✓	✓
Chess	✓	✗	✓	✗	✓
Endgames	✓	✗	✓	✗	✓

Figure 2: We have designed our RL Capability tasks as text games that include Chess, Endgames, Wordle, Maze, and Text-Nav. These tasks isolate some subset of the RL Capabilities outlined in Appendix B.1.

Desirable RL capabilities. RL shines in goal-directed tasks that require multi-step planning and *strategic decision-making*. Strategic decision-making can range from asking follow-up questions (e.g. 20 Questions), to complex strategy in chess. In RL, it is necessary that algorithms can properly perform *credit assignment* as rewards are often delayed relative to the action pivotal to the outcome. A challenge with optimizing POMDPs is *partial observability*, where the agent must make deductions based on incomplete information. In the offline RL setting, the ability of algorithms to perform *trajectory stitching* is often desirable for learning optimal policies from suboptimal trajectories. Lastly, when working with language models, it's important that algorithms remain effective in the face of *complex language* with open-ended generation. We design our RL-capability tests with the goal of stress-testing each of these capabilities, as shown in Figure 2.

Maze and Text-Nav. We consider a Maze task as well as the Text-Nav featuring more complex language. Though Text-Nav involves stochastic language, the maze task has longer dataset trajectories and a more complicated layout. To test partial observability, we include both a partially observed and fully observed version of each task. In the partially observed version, we remove information from the maze description such that the agent must infer its position from its move history. To emphasize the comparison to a non-text-based version, we evaluate the Maze task in a symbolic or grid-based environment seen in Appendix H.

Strategy games. We include three strategy games; Wordle, Chess, and Endgames. Wordle tests partial observability over the space of possible words while Chess and Endgames test the ability of the agent to form longer-term plans. Endgames provide a simpler and more goal-directed variation of the Chess task. By focusing on the endgame, we encourage algorithms to learn strategy rather than memorizing the opening moves of a chess game. A classic theoretical endgame position consists of a position where the only pieces on the board are the two kings and the queen. All RL Capability tasks evaluate *trajectory stitching* capability through the inclusion of suboptimal trajectories. Further details about our dataset generation strategies can be found in Appendix D. The Chess, Endgames, Maze and Text-Nav tasks test *credit assignment*, because the RL algorithm must learn to assign credit to good actions rather than a lucky starting position in the maze task, or a weak opponent moves in the Chess or Endgames task.

4.3 AN OVERVIEW OF DATA COLLECTION FOR LMRL-GYM

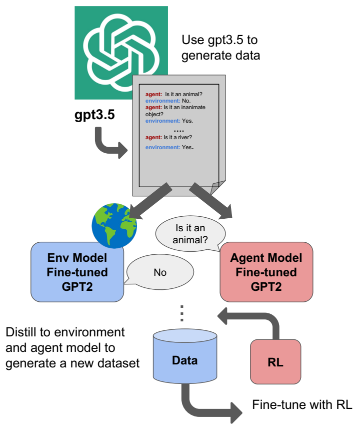

Figure 3: To generate data for conversational tasks, we use LLMs as "simulators" for the task. Our simulators can be used to generate offline data, to provide a "simulation environment" for evaluation, to perform online training, and to compute rewards.

To make tasks in LMRL-Gym practical for benchmarking RL methods, we must balance accessibility and realism. As RL algorithms need to be evaluated by running a learned policy, real-world tasks are comparatively inaccessible for rapid iteration (e.g., if they require talking to real humans). We therefore use simulators for our tasks, derived either from text-based games, or conversational agents powered by language models. Although this fully synthetic setup sacrifices the realistic nature of tasks, we believe significant gain in accessibility is worthwhile and will enable rapid RL algorithm progress.

RL Capability tests. For each task, we use a simulator such as a chess engine or maze solver to generate near-optimal data and then we dilute the policy with suboptimal data by taking suboptimal actions or using inferior policies. We also convert our task from a symbolic version to a text-based version in a programmatic way as discussed in Appendix B.

Interactive Dialogue tasks. For conversational tasks, we leverage existing LLMs to generate our data, either with two instances of LLMs "talking" to one another or all at once through few-shot prompting as shown in Figure 3. To train these LLMs, we use OpenAI's GPT-3.5 to generate an initial dataset by asking reasonable questions and answers out-of-the-box, collecting a dataset of differing sizes depending on the task. In the case of 20Qs and Guess My City, we collected 1K conversations by querying GPT-3.5 (text-davinci-003) to generate both sides of the conversation based on specific prompts (which can be found in Appendix D.6. To generate the dataset for training our algorithms, we fine-tuned a FLAN-T5-XL guesser model and a FLAN-T5-XL oracle model on their respective sides of the conversation. Using these distilled models, we generated a new dataset of 100K conversations by having the two models talk to each other. We conducted a similar process for the Car Dealer task but with a larger model for fine-tuning (GPT2-XL). When generating our datasets, we also spent considerable effort to ensure diversity in the responses to ensure the collection of high-quality data. For the Car Dealer task as an example, this included providing different desired brands, features, classifications (i.e. car or truck), and budgets in our prompting to generate the datasets. Further details on our data generation process for the three Interactive Dialogue tasks can be found in Appendix D.

Task	`20Qs`	`Guess`	`Car`	`Maze`	`Text-Nav`	`Wordle`	`Chess`	`Endgames`
`Size`	100k	100k	19k	1.24k	2.5k	1m	625k	97.756k
`avg length`	14.9	18.8	16.5	19.7	12.2	4.82	46.7	11.9
`std length`	4.38	4.57	3.61	24.5	8.77	1.27	18.16	12.0
`success rate`	0.31	0.53	0.53	0.11	0.26	0.70	0.60	0.59
`avg return`	-17.3	-18.8	0.562	-19.7	0.258	-4.12	0.210	0.586
`std return`	2.56	4.12	0.422	24.5	0.424	1.59	0.970	0.492

Table 1: Statistics for all tasks in LMRL-Gym. Size represents the number of trajectories, the average length is the average length of trajectories in the dataset where the unit is a response from the agent. The success rate is the proportion of trajectories that reach the objective. Finally, the reward functions for each task are defined in Appendix D.

5 LMRL-Gym Research Framework for Algorithm Development

We evaluate the LMRL-Gym tasks on both online and offline RL algorithms, including variations of behavior cloning, value-based RL methods, and online PPO. We have selected these algorithms have they are currently the state-of-the-art methods RL methods for LLMs Chen et al. (2021a); Snell et al. (2022a); Ouyang et al. (2022). With these experiments, we expect to observe (1) a significant spread in performance between the different algorithms, highlighting differences between RL algorithms; (2)

room to improve beyond, such that our benchmark can enable future algorithmic development. Our project page (REDACTED) contains links to our open-sourced datasets (REDACTED) and research framework (REDACTED).

BC, Filtered BC, Online Filtered BC. In line with standard RL nomenclature, we denote supervised fine-tuning as behavioral cloning (BC). This baseline tests whether LMs can effectively represent the behaviors in the datasets. Filtered BC is identical, except only the most successful examples in the offline dataset are used for fine-tuning, a technique which is also used in Snell et al. (2022a). Online filtered BC collects data online using the current policy and selects the most successful trajectories for finetuning. See Appendix E for our data filtering criteria for each task.

Offline Value-based RL: MC Returns and ILQL. Monte-Carlo returns (Kakutani, 1945) and Implicit Language Q-Learning (Snell et al., 2022a) train a value V and Q function. In MC Returns, we train the Q function with an MSE to predict the reward-to-go. In ILQL we train the two action-value (Q) functions using the Bellman backup operator (Kostrikov et al., 2021). For both algorithms, the Q and V functions are then used to perturb the logits of the original BC model (see Equation 5).

Online RL: PPO. PPO (Schulman et al., 2017) is an online RL algorithm widely adopted for training language models with Reinforcement Learning from Human Feedback (Christiano et al., 2023; Stiennon et al., 2022; Bai et al., 2022a; Casper et al., 2023). Unlike previous value-function RL methods, PPO learns a language model policy with no policy extraction step.

GPT4. Few-shot prompting is a common technique for creating interactive language agents Wang et al. (2023a). To compare this to RL fine-tuning we few-shot prompt GPT4 using dataset examples and a detailed explanation of the game for each task. The prompts can be found in our code repository.

Training and evaluation protocol for algorithms. For the BC and filtered BC methods, we initialize our models with the pre-trained GPT2 weights (Radford et al., 2019) and perform standard fine-tuning. We choose GPT2 rather than a larger model due to memory and time constraints, though we admit larger models would lead to a performance boost. For each of the RL methods, we initialize the weights of the base model with the weights from the BC checkpoint and then continue finetuning with the RL objective. When fine-tuning PPO, we limit the number of samples to less than 100k. We report the hyperparameters that we used for each task in Appendix E. We evaluate each policy by measuring the average reward in the simulated environment for each task.

Evaluation of data generation. When using LLMs as a simulator for human actions, it is important to verify that (1) the text produced by the LLM is natural and (2) LLM simulator is not exploitable e.g. policy achieves high reward without actually accomplishing the goal. In addition to validating the data generation process through statistics reported in Table 1, we verified the naturalness of the LLM-produced text in a user study of 40 users. In this study, found no significant difference in the naturalness of conversations generated by ChatGPT3.5 and our trained simulators and agents Appendix A. For example, natural conversations imply that the strategies employed by the Seller to convince the Buyer followed human patterns of conversation and indicate the robustness of the Buyer model to hacking. 20 Questions and Guess My City are particularly hard to hack as they require the agent to successfully guess the word. We verify this through automatic checks as described in our prompting strategy in Appendix D.6.

6 Benchmarking Baseline RL Methods

In Table 2 we present the results for each method on each of our text-game and interactive dialogue tasks. We normalize the scores such that a score of 50 corresponds to the average reward in our offline dataset, 0 corresponds to the lowest possible score, and 100 to the highest score. Across all tasks, we see that our offline RL baseline methods consistently outperform both the dataset and the filtered BC policies, demonstrating the efficacy of offline RL in representing a more optimal policy than the best behaviors in the data. Similarly, we see that online PPO generally improves over the BC policies, highlighting the utility of learning from online environment interaction. However, between RL Capability tasks and Interactive Dialogue tasks, we observe desperate trends in which specific method performs the best. We discuss this in more detail below.

	alg.	BC	% BC	MC Return	ILQL	Online PPO	Online % BC	GPT-4
Interactive Dialogue	`20Qs`	57.1	77.1	87.1	82.9	72.9	55.2	**95.7**
	`Guess`	30.0	48.0	88.0	75.0	49.9	31.6	**92.3**
	`Car`	44.5	54.8	**57.2**	46.3	50.5	40.4	53.5
RL Capability tasks	`FO Maze`	58.2	68.9	75.0	**99.9**	79.7	57.4	78.2
	`PO Maze`	53.1	50.1	52.4	**76.3**	42.4	53.1	60.4
	`FO Text-Nav`	53.7	65.1	71.9	**91.8**	87.1	74.5	67.5
	`PO Text-Nav`	49.7	60.5	71.6	83.7	**85.5**	68.4	40.2
	`Wordle`	79.9	79.1	94.9	**97.7**	84.2	95.2	15.4
	`Chess`	47.2	42.9	46.5	47.3	**48.0**	47.2	0
	`Endgames`	35.1	17.7	50.2	45.8	**77.5**	36.2	0

Table 2: Normalized reward for all tasks. We present the interactive dialogue tasks on top and the RL capability tasks on the bottom. Value-based methods (MC and ILQL) generally outperform filtered BC, as we might expect in stochastic settings, though the relative performance of ILQL and the simpler MC method is, perhaps surprisingly, reversed on the tasks with more complex language, suggesting that there is room for improvement with such methods. Online RL with PPO often, but not always, improves over offline methods that are not permitted to collect additional online interaction. To make the results more comparable across tasks, we normalize the average return for each policy such that 0 is the minimum possible return, 50 is the dataset average return, and 100 is the maximum return for each task. We also report the raw score results and evaluation details in Appendix F.

Which algorithm performs best on the RL Capability tasks? On the RL Capability tasks in Table 2, we see ILQL has the highest performance across all methods for most tasks. ILQL's performance on these tasks is likely due to its unique ability to perform trajectory stitching, enabling it to outperform any individual trajectory in the dataset by learning to compose the best parts of many different trajectories. However, on the PO text-nav, chess, and endgames tasks, we see that PPO outperforms ILQL, suggesting that there is likely still much room for improvement in terms of developing better offline TD-based RL methods for LLMs.

Which offline RL algorithm performs best for Interactive Dialouge tasks? In contrast to the text-based games, on our Interactive Dialogue tasks, we see that across all tasks ILQL under-performs the simpler MC returns method. This discrepancy with dialogue, may be because on the more complex text-based tasks it is harder to scale full TD-learning. In fact, we find that on the car-dealer task, even filtered BC outperforms ILQL. Overall, these findings demonstrate that there is much progress to be made in developing better offline RL methods that can effectively optimize LLMs in complex and realistic dialogue settings.

How does performance of language-based text games compare with their symbolic-based counterparts? We created a non-text-based version of the Maze task (an RL Capability task) to investigate what difficulties arise from deploying RL algorithms on language-based tasks. We found that simple online and offline Q-learning was able to get an optimal score on the maze. Therefore, the performance symbolic maze is comparable to the fully observed Maze task. However, on the PO Maze task, the language-based methods perform significantly worse. This highlights room for improvement in dealing with partial observability in environments with complex language. Further details for this ablation are found in Appendix H.

How does prompting GPT-4 compare with RL fine-tuning? On the RL Capability tasks, we find that our much smaller RL finetuned models significantly outperform GPT4, demonstrating the efficacy of RL for enabling complex goal-directed behaviors in language models. However, on the Interactive Dialogue tasks, GPT-4 outperforms or performs on par with our best RL-trained models. These dialogue tasks are likely to be much more in distribution for GPT4 than our text-game RL capability tasks, and thus GPT4's broad world-knowledge, reasoning, and conversational abilities become synchronized allowing it to compensate for its lack of goal-directed RL fine-tuning in these scenarios. Nonetheless, the mere fact that finetuning small models with RL enables us to close much of the gap to GPT4 on these more realistic tasks underscores the efficacy of RL finetuning. In summary, we can see that RL algorithms consistently outperform baselines like filtered BC on many of the tasks. However, these results highlight significant areas for growth. For example, the instabilities observed in training PPO require further investigation beyond hyperparameter tuning. Moreover, the performance discrepancy between ILQL and the simpler MC Returns highlights that scaling full TD-learning to Interactive Dialogue settings is another area for improvement.

7 Discussion

We propose LMRL-Gym, consisting of 8 tasks including three Interactive Dialogue tasks, and five RL Capability tests. We provide a research toolkit for practitioners to get started with multi-turn RL for LLMs. Our objective is enable the iteration and development of more effective methods for language-based, multi-turn interaction tasks. This includes enabling core capabilities in LLMs through RL to perform complex decision-making, complex conversational interactions, credit assignment, and trajectory stitching. Our evaluation shows promise of RL in several tasks, with further room for improvement with a push for better methods. We acknowledge several limitations when designing tasks in our benchmark, including primarily leveraging smaller GPT-based LLMs to generate datasets and finetune our LLM-based simulators. While we have primarily trained and evaluated models with a maximum 1.5B parameters, we have maintained a lower parameter count to ensure accessibility for researchers with limited computational resources. In addition to releasing our code and datasets, we share all of the hyperparameters we used to train our models in Appendix E and provide more in-depth insight into our results, training procedure, and evaluation in Appendix F.

We would like to acknowledge that this work is part of a larger effort to improve the performance of LLMs in settings that require planning or multi-turn interactions including multi-turn dialogue, complex tool use, multi-step games, and other interactive applications. Our goal is to propose tasks to evaluate different capabilities expected from an LLM, such as common sense reasoning, credit assignment, reasoning under uncertainty, information-seeking behaviors, and trajectory stitching. We hope this benchmark inspires the creation of more synthetic datasets and simulators for dialogue and is used to design better algorithms to train goal-directed LLM-RL models.

8 Impact Statement

This work aims to develop a benchmark for the advancement of research in reinforcement learning and LLMs. We generate datasets for tasks in our benchmark with existing LLMs for dialogue tasks and online engines for text games, adhering to best practices in data handling and ensuring there is no personally identifiable or sensitive information present in the generated datasets. We recognize that there may be biases present in the datasets we collect, and have taken steps to ensure a diverse and varied collection of responses from LLMs for our conversational task as detailed in our data generation process in Appendix D. In considering the ethical implications of interactive RL, we acknowledge the dual use implication of this research, particularly centered around developing LLM simulators that could perform persuasion, manipulation, and addictive engagement of users at a large scale. The optimization processes employed by such algorithms, which aim to maximize certain objectives, raise ethical considerations when the optimized outcomes may prioritize system goals over user safety and alignment to human values. We have designed our datasets and reward functions such that prioritize fairness and human-aligned outcomes. By incorporating these considerations when designing our framework, we aim to encourage the development of reinforcement learning models and LLMs that not only excel in performance but also adhere to ethical standards.

References

Prithviraj Ammanabrolu, Ethan Tien, Matthew Hausknecht, and Mark O Riedl. How to avoid being eaten by a grue: Structured exploration strategies for textual worlds. *arXiv preprint arXiv:2006.07409*, 2020.

Anthropic. Introducing claude, 2023. URL `https://www.anthropic.com/index/introducing-claude`.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan

Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022a.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosuite, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemi Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. Constitutional ai: Harmlessness from ai feedback, 2022b.

Florian Böhm, Yang Gao, Christian M. Meyer, Ori Shapira, Ido Dagan, and Iryna Gurevych. Better rewards yield better summaries: Learning to summarise without references. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3110–3120, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1307. URL https://aclanthology.org/D19-1307.

Ond rej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pp. 131–198, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W/W16/W16-2301.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.

Thomas Carta, Clément Romac, Thomas Wolf, Sylvain Lamprier, Olivier Sigaud, and Pierre-Yves Oudeyer. Grounding large language models in interactive environments with online reinforcement learning, 2023.

Stephen Casper, Xander Davies, Claudia Shi, Thomas Krendl Gilbert, Jérémy Scheurer, Javier Rando, Rachel Freedman, Tomasz Korbak, David Lindner, Pedro Freire, Tony Wang, Samuel Marks, Charbel-Raphaël Segerie, Micah Carroll, Andi Peng, Phillip Christoffersen, Mehul Damani, Stewart Slocum, Usman Anwar, Anand Siththaranjan, Max Nadeau, Eric J. Michaud, Jacob Pfau, Dmitrii Krasheninnikov, Xin Chen, Lauro Langosco, Peter Hase, Erdem Bıyık, Anca Dragan, David Krueger, Dorsa Sadigh, and Dylan Hadfield-Menell. Open problems and fundamental limitations of reinforcement learning from human feedback, 2023.

Louis Castricato, Alex Havrilla, Shahbuland Matiana, Duy V. Phung, Aman Tiwari, Jonathan Tow, and Maksym Zhuravinsky. trlX: A scalable framework for RLHF, June 2023. URL https://github.com/CarperAI/trlx.

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

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, Alex Ray, Raul Puri, Gretchen Krueger, Michael Petrov, Heidy Khlaaf, Girish Sastry, Pamela Mishkin, Brooke Chan, Scott Gray, Nick Ryder, Mikhail Pavlov, Alethea Power, Lukasz Kaiser, Mohammad Bavarian, Clemens Winter, Philippe Tillet, Felipe Petroski Such, Dave Cummings, Matthias Plappert, Fotios

Chantzis, Elizabeth Barnes, Ariel Herbert-Voss, William Hebgen Guss, Alex Nichol, Alex Paino, Nikolas Tezak, Jie Tang, Igor Babuschkin, Suchir Balaji, Shantanu Jain, William Saunders, Christopher Hesse, Andrew N. Carr, Jan Leike, Josh Achiam, Vedant Misra, Evan Morikawa, Alec Radford, Matthew Knight, Miles Brundage, Mira Murati, Katie Mayer, Peter Welinder, Bob McGrew, Dario Amodei, Sam McCandlish, Ilya Sutskever, and Wojciech Zaremba. Evaluating large language models trained on code, 2021b.

Maxime Chevalier-Boisvert, Dzmitry Bahdanau, Salem Lahlou, Lucas Willems, Chitwan Saharia, Thien Huu Nguyen, and Yoshua Bengio. Babyai: A platform to study the sample efficiency of grounded language learning. *arXiv preprint arXiv:1810.08272*, 2018.

Paul Christiano, Jan Leike, Tom B. Brown, Miljan Martic, Shane Legg, and Dario Amodei. Deep reinforcement learning from human preferences, 2023.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Marc-Alexandre Côté, Ákos Kádár, Xingdi Yuan, Ben Kybartas, Tavian Barnes, Emery Fine, James Moore, Matthew J. Hausknecht, Layla El Asri, Mahmoud Adada, Wendy Tay, and Adam Trischler. Textworld: A learning environment for text-based games. *CoRR*, abs/1806.11532, 2018.

Heriberto Cuayáhuitl, Simon Keizer, and Oliver Lemon. Strategic dialogue management via deep reinforcement learning, 2015.

Maxime De Bruyn, Ehsan Lotfi, Jeska Buhmann, and Walter Daelemans. 20q: Overlap-free world knowledge benchmark for language models. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pp. 494–508, 2022a.

Maxime De Bruyn, Ehsan Lotfi, Jeska Buhmann, and Walter Daelemans. Is it smaller than a tennis ball? language models play the game of twenty questions. In *Proceedings of the Fifth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pp. 80–90, 2022b.

Meta Fundamental AI Research Diplomacy Team (FAIR)†, Anton Bakhtin, Noam Brown, Emily Dinan, Gabriele Farina, Colin Flaherty, Daniel Fried, Andrew Goff, Jonathan Gray, Hengyuan Hu, Athul Paul Jacob, Mojtaba Komeili, Karthik Konath, Minae Kwon, Adam Lerer, Mike Lewis, Alexander H. Miller, Sasha Mitts, Adithya Renduchintala, Stephen Roller, Dirk Rowe, Weiyan Shi, Joe Spisak, Alexander Wei, David Wu, Hugh Zhang, and Markus Zijlstra. Human-level play in the game of <i>diplomacy</i> by combining language models with strategic reasoning. *Science*, 378(6624):1067–1074, 2022. doi: 10.1126/science.ade9097. URL https://www.science.org/doi/abs/10.1126/science.ade9097.

Angela Fan, Jack Urbanek, Pratik Ringshia, Emily Dinan, Emma Qian, Siddharth Karamcheti, Shrimai Prabhumoye, Douwe Kiela, Tim Rocktaschel, Arthur Szlam, et al. Generating interactive worlds with text. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 1693–1700, 2020.

Kallirroi Georgila and David Traum. Reinforcement learning of argumentation dialogue policies in negotiation. pp. 2073–2076, 08 2011. doi: 10.21437/Interspeech.2011-544.

Marjan Ghazvininejad, Xing Shi, Jay Priyadarshi, and Kevin Knight. Hafez: an interactive poetry generation system. In *Proceedings of ACL 2017, System Demonstrations*, pp. 43–48, Vancouver, Canada, July 2017. Association for Computational Linguistics. URL https://aclanthology.org/P17-4008.

Nicolas Gontier, Pau Rodriguez, Issam Laradji, David Vazquez, and Christopher Pal. Language decision transformers with exponential tilt for interactive text environments. *arXiv preprint arXiv:2302.05507*, 2023.

Google. Bard, 2023. URL https://bard.google.com/.

Jiatao Gu, Kyunghyun Cho, and Victor O.K. Li. Trainable greedy decoding for neural machine translation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 1968–1978, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1210. URL https://aclanthology.org/D17-1210.

Xiaoxiao Guo, Mo Yu, Yupeng Gao, Chuang Gan, Murray Campbell, and Shiyu Chang. Interactive fiction game playing as multi-paragraph reading comprehension with reinforcement learning. *arXiv preprint arXiv:2010.02386*, 2020.

Matthew Hausknecht, Prithviraj Ammanabrolu, Côté Marc-Alexandre, and Yuan Xingdi. Interactive fiction games: A colossal adventure. *CoRR*, abs/1909.05398, 2019. URL http://arxiv.org/abs/1909.05398.

Matthew Hausknecht, Prithviraj Ammanabrolu, Marc-Alexandre Côté, and Xingdi Yuan. Interactive fiction games: A colossal adventure. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 7903–7910, 2020.

He He, Derek Chen, Anusha Balakrishnan, and Percy Liang. Decoupling strategy and generation in negotiation dialogues, 2018.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*, 2021a.

Dan Hendrycks, Mantas Mazeika, Andy Zou, Sahil Patel, Christine Zhu, Jesus Navarro, Dawn Song, Bo Li, and Jacob Steinhardt. What would jiminy cricket do? towards agents that behave morally. *NeurIPS*, 2021b.

Robert Irvine, Douglas Boubert, Vyas Raina, Adian Liusie, Vineet Mudupalli, Aliaksei Korshuk, Zongyi Liu, Fritz Cremer, Valentin Assassi, Christie-Carol Beauchamp, et al. Rewarding chatbots for real-world engagement with millions of users. *arXiv preprint arXiv:2303.06135*, 2023.

Youngsoo Jang, Jongmin Lee, and Kee-Eung Kim. GPT-critic: Offline reinforcement learning for end-to-end task-oriented dialogue systems. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=qaxhBG1UUaS.

Peter A Jansen and Marc-Alexandre Côté. Textworldexpress: Simulating text games at one million steps per second. *arXiv preprint arXiv:2208.01174*, 2022.

N. Jaques, J. H. Shen, A. Ghandeharioun, C. Ferguson, A. Lapedriza, N. Jones, S. Gu, and R. Picard. Human-centric dialog training via offline reinforcement learning. *Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

Natasha Jaques, Shixiang Gu, Dzmitry Bahdanau, José Miguel Hernández-Lobato, Richard E. Turner, and Douglas Eck. Sequence tutor: Conservative fine-tuning of sequence generation models with KL-control. In Doina Precup and Yee Whye Teh (eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1645–1654. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/jaques17a.html.

Shizuo Kakutani. Markoff process and the dirichlet problem. *Proceedings of the Japan Academy*, 21 (3-10):227–233, 1945.

Samuel Kiegeland and Julia Kreutzer. Revisiting the weaknesses of reinforcement learning for neural machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1673–1681, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.133. URL https://aclanthology.org/2021.naacl-main.133.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169*, 2021.

Wai-Chung Kwan, Hongru Wang, Huimin Wang, and Kam-Fai Wong. A survey on recent advances and challenges in reinforcement learning methods for task-oriented dialogue policy learning, 2022.

Mike Lewis, Denis Yarats, Yann N. Dauphin, Devi Parikh, and Dhruv Batra. Deal or no deal? end-to-end learning for negotiation dialogues, 2017.

Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. Deep reinforcement learning for dialogue generation, 2016.

Mitchell P Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. Building a large annotated corpus of english: The penn treebank. *Computational Linguistics*, 19(2):313–330, 1993.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. Cross-task generalization via natural language crowdsourcing instructions. In *ACL*, 2022.

Karthik Narasimhan, Tejas Kulkarni, and Regina Barzilay. Language understanding for text-based games using deep reinforcement learning. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1–11, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1001. URL https://aclanthology.org/D15-1001.

Khanh Nguyen, Hal Daumé III, and Jordan Boyd-Graber. Reinforcement learning for bandit neural machine translation with simulated human feedback. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 1464–1474, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1153. URL https://aclanthology.org/D17-1153.

OpenAI. Chatgpt, 2022. URL https://openai.com/blog/chatgpt.

OpenAI. Gpt-4, 2023. URL https://openai.com/research/gpt-4.

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

Richard Yuanzhe Pang and He He. Text generation by learning from demonstrations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=RovX-uQ1Hua.

Joon Sung Park, Joseph C. O'Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. Generative agents: Interactive simulacra of human behavior, 2023.

Romain Paulus, Caiming Xiong, and Richard Socher. A deep reinforced model for abstractive summarization, 2017.

Valentina Pyatkin, Jena D. Hwang, Vivek Srikumar, Ximing Lu, Liwei Jiang, Yejin Choi, and Chandra Bhagavatula. Reinforced clarification question generation with defeasibility rewards for disambiguating social and moral situations, 2022.

Yujia Qin, Shengding Hu, Yankai Lin, Weize Chen, Ning Ding, Ganqu Cui, Zheni Zeng, Yufei Huang, Chaojun Xiao, Chi Han, et al. Tool learning with foundation models. *arXiv preprint arXiv:2304.08354*, 2023.

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

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

Rajkumar Ramamurthy, Prithviraj Ammanabrolu, Kianté Brantley, Jack Hessel, Rafet Sifa, Christian Bauckhage, Hannaneh Hajishirzi, and Yejin Choi. Is reinforcement learning (not) for natural language processing: Benchmarks, baselines, and building blocks for natural language policy optimization. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=8aHzds2uUyB.

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

John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation, 2018.

Pararth Shah, Dilek Hakkani-Tur, and Larry Heck. Interactive reinforcement learning for task-oriented dialogue management. 2016.

Kurt Shuster, Mojtaba Komeili, Leonard Adolphs, Stephen Roller, Arthur Szlam, and Jason Weston. Language models that seek for knowledge: Modular search & generation for dialogue and prompt completion, 2022a.

Kurt Shuster, Jing Xu, Mojtaba Komeili, Da Ju, Eric Michael Smith, Stephen Roller, Megan Ung, Moya Chen, Kushal Arora, Joshua Lane, Morteza Behrooz, William Ngan, Spencer Poff, Naman Goyal, Arthur Szlam, Y-Lan Boureau, Melanie Kambadur, and Browser assisted question-answering with human feedbackJason Weston. Blenderbot 3: a deployed conversational agent that continually learns to responsibly engage, 2022b.

Ishika Singh, Gargi Singh, and Ashutosh Modi. Pre-trained language models as prior knowledge for playing text-based games. *arXiv preprint arXiv:2107.08408*, 2021.

Satinder Singh, Michael Kearns, Diane Litman, and Marilyn Walker. Reinforcement learning for spoken dialogue systems. *Advances in neural information processing systems*, 12, 1999.

Charlie Snell, Ilya Kostrikov, Yi Su, Mengjiao Yang, and Sergey Levine. Offline rl for natural language generation with implicit language q learning. *arXiv preprint arXiv:2206.11871*, 2022a.

Charlie Snell, Sherry Yang, Justin Fu, Yi Su, and Sergey Levine. Context-aware language modeling for goal-oriented dialogue systems. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pp. 2351–2366, Seattle, United States, July 2022b. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-naacl.181. URL https://aclanthology.org/2022.findings-naacl.181.

Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.

Nisan Stiennon, Long Ouyang, Jeff Wu, Daniel M. Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul Christiano. Learning to summarize from human feedback, 2022.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Pradyumna Tambwekar, Murtaza Dhuliawala, Lara J. Martin, Animesh Mehta, Brent Harrison, and Mark O. Riedl. Controllable neural story plot generation via reward shaping. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, aug 2019. doi: 10.24963/ijcai.2019/829. URL https://doi.org/10.24963%2Fijcai.2019%2F829.

Jörg Tiedemann. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pp. 2214–2218, Istanbul, Turkey, May 2012. European Language Resources Association (ELRA). URL http://www.lrec-conf.org/proceedings/lrec2012/pdf/463_Paper.pdf.

Erik F Tjong Kim Sang and Fien De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003*, volume 4, pp. 142–147, 2003.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023.

Siddharth Verma, Justin Fu, Mengjiao Yang, and Sergey Levine. Chai: A chatbot ai for task-oriented dialogue with offline reinforcement learning, 2022.

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. Tl; dr: Mining reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pp. 59–63, 2017.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rJ4km2R5t7.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*, 2023a.

Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. Scienceworld: Is your agent smarter than a 5th grader? *arXiv preprint arXiv:2203.07540*, 2022a.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, et al. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pp. 5085–5109, 2022b.

Yue Wang, Hung Le, Akhilesh Deepak Gotmare, Nghi D. Q. Bui, Junnan Li, and Steven C. H. Hoi. Codet5+: Open code large language models for code understanding and generation, 2023b.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. Finetuned language models are zero-shot learners, 2022.

Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. *arXiv preprint arXiv:1804.08198*, 2018.

Jeff Wu, Long Ouyang, Daniel M Ziegler, Nisan Stiennon, Ryan Lowe, Jan Leike, and Paul Christiano. Recursively summarizing books with human feedback. *arXiv preprint arXiv:2109.10862*, 2021.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation, 2016.

Yuxiang Wu and Baotian Hu. Learning to extract coherent summary via deep reinforcement learning, 2018.

Shunyu Yao, Rohan Rao, Matthew Hausknecht, and Karthik Narasimhan. Keep calm and explore: Language models for action generation in text-based games. *arXiv preprint arXiv:2010.02903*, 2020.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*, 2022.

Shunyu Yao, Howard Chen, John Yang, and Karthik Narasimhan. Webshop: Towards scalable real-world web interaction with grounded language agents, 2023.

Xingdi Yuan, Jie Fu, Marc-Alexandre Cote, Yi Tay, Christopher Pal, and Adam Trischler. Interactive machine comprehension with information seeking agents. *arXiv preprint arXiv:1908.10449*, 2019.

Yizhe Zhang, Jiarui Lu, and Navdeep Jaitly. The entity-deduction arena: A playground for probing the conversational reasoning and planning capabilities of llms, 2023.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences, 2020.