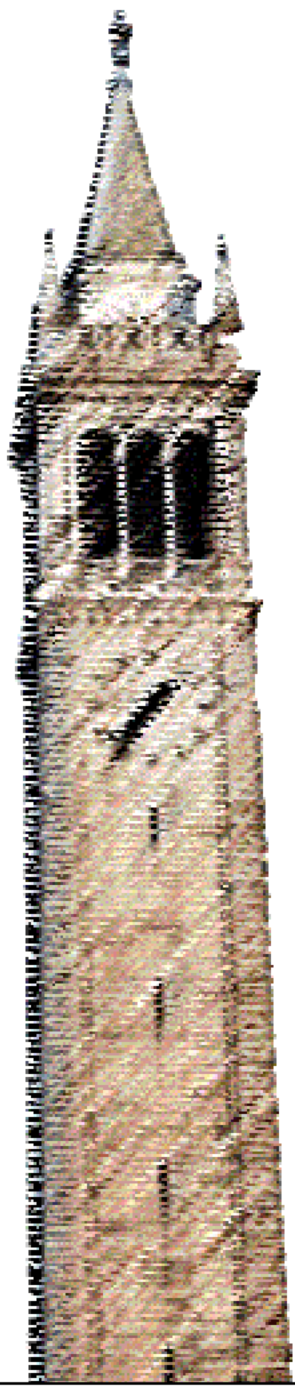
Benchmarks for RL on Goal-directed Language Tasks with LLMs



Charles Sun

Electrical Engineering and Computer Sciences
University of California, Berkeley

Technical Report No. UCB/EECS-2023-98
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2023/EECS-2023-98.html

May 11, 2023

Acknowledgement

I would like to thank my research advisor, Professor Sergey Levine, for his support throughout my academic career. This work is part of a larger joint effort with Charlie Snell, Marwa Abdulhai, Simon Zhai, and Isadora White, and would not have been possible without them. I would also like to thank my family and friends for their continued support.

Benchmarks for RL on Goal-directed Language Tasks with LLMs

Charles Sun

Research Project

Submitted to the Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, in partial satisfaction of the requirements for the degree of **Master of Science, Plan II**.

Approval for the Report and Comprehensive Examination:

Committee

Professor Sergey Levine
Research Advisor

5/10/23

(Date)

⋆ ⋆ ⋆ ⋆ ⋆ ⋆ ⋆

Professor Pieter Abbeel
Second Reader

5/10/23

(Date)

Abstract

Large language models (LLMs) have demonstrated remarkable abilities when conversing with humans, answering questions, and responding to requests. Much of this capability is enabled through learning from large datasets taken from the Internet and finetuning on human preferences with RLHF. However, LLMs trained this way are not explicitly long-term goal-directed, as they are not optimized with an explicitly defined long-term objective. Reinforcement learning (RL) aims to solve the long-term goal-directed problem, and has been extremely successful on a wide variety of non-language tasks. However, progress for RL on goal-directed language tasks with LLMs has been lacking. A major roadblock with leveraging RL for goal-directed language tasks is the lack of clarity with respect to the tasks that it is best suited towards. We propose LLM-RL, a diverse suite of tasks and a set of corresponding datasets that will allow us to illustrate the potential of RL algorithms in goal-directed language tasks.

Acknowledgement

I would like to thank my research advisor, Professor Sergey Levine, for his support throughout my academic career. This work is part of a larger joint effort with Charlie Snell, Marwa Abdulhai, Simon Zhai, and Isadora White, and would not have been possible without them. I would also like to thank my family and friends for their continued support.

Contents

1 Introduction 5
2 Related Works 7
3 Background 9
4 Tasks and Datasets 10
4.1 Twenty Questions 11
4.2 Car Dealer 12
5 Benchmarking Prior Methods 18
5.1 Twenty Questions 20
5.2 Car Dealer 21
6 Discussions 23
Bibliography 24
Appendix 31
1 Twenty Questions 31
1.1 List of Words 31
1.2 Prompt Details 32
1.3 Examples 34
2 Car Dealer 36
2.1 Examples 36

1 Introduction

Large language models (LLMs) have demonstrated remarkable abilities when naturally conversing with humans [39, 40], answering questions and responding to requests [50, 51], and even performing coding tasks [11]. Much of this capability is enabled by learning to emulate humans from large datasets of text from the web [50, 61, 71], learning from examples "in context" [8, 66], as well as other sources of supervision such as instruction datasets [35, 64, 65] and preference fine-tuning with RLHF [41, 73]. LLMs trained in this way are not explicitly goal-directed, as they are not optimized to directly solve particular tasks or optimize an explicitly defined objective, but rather to produce text that resembles the distribution of human-provided examples or accords with human preferences [3, 55, 67, 73]. This can become particularly apparent in the context of temporally extended tasks, such as multi-turn dialogue [1, 25], complex tool use [63], multi-step games [23], as well as 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 planning and optimization. Multi-turn reinforcement learning (RL) [56] offers a path to enable LLMs to overcome challenges in goal-directed reasoning and planning in interactive, multi-turn settings, including complex dialogue, games, and tool use.

With the empirical successes in large-scale sequential decision-making in other domains [2, 5, 52, 60], we might suppose that RL could serve as a powerful tool for LLM training, not just for training models in accordance with human preferences, but more generally to accomplish tasks in an intentional and goal-directed manner. Text generation can naturally be viewed as a sequential decision-making process by treating a sequence of tokens as a trajectory. As RL aims at training intelligent agents by maximizing the overall sum of rewards in one trajectory, RL is expected to improve the performance of goal-directed LLMs tasks such as dialogue, text games, or tool use. Learning with RL has generally improved the capabilities afforded by language models such as question answering, where RL-tuned LLMs have been shown to better understand complex questions and provide more accurate and relevant answers through step-by-step reasoning and human feedback. Applied to dialogue settings, such as in ChatGPT [39], the RLHF approaches yield more effective conversational agents by incorporating human feedback such as user satisfaction and task completion, generating more natural and effective responses to user queries and feedback. However, despite the increasing interest in RL research for LMs [9, 41, 48, 53, 55], current LLM benchmarks either only focus one single-step or non-goal-direct tasks [13, 21, 22, 47, 48, 62], or lack an easy and efficient method for evaluating success despite being multi-step and goal-directed [31, 59].

The full potential of RL with LLMs would be unlocked if we can develop methods that can optimize not just text generation, but *interaction*, enabling LLMs to learn from actual interactions with humans, tools, and games. To this end, we introduce the benchmark LLM-RL, for goal-directed multi-step reinforcement learning with LLMs. Our benchmark consists of two different language tasks that require multiple rounds of language interaction and cover a range of tasks in open-ended dialogue and text games. We provide environment simulators as well as datasets to facilitate RL research. We also provide detailed descriptions of why we chose our tasks and the data collection process for each task, which could enable researchers in the future to collect more data for our tasks, or to propose different goal-directed multi-step LLM RL tasks. Finally, we provide well-defined reward functions for each of our tasks and evaluate performances of current state-of-the-art supervised learning and RL algorithms on each of our tasks as baselines. Our full datasets can be found here: `https://github.com/charlesjsun/benchmarks-llm-rl`.

2 Related Works

Datasets, Benchmarks, and Libraries. There have been a series of benchmarks for machine translation [7, 58], natural language understanding [21, 22, 47, 48, 62], and solving math problems [13]. However, these are not multi-step decision making tasks. There have been a few dialogue-based sequential datasets such as CraiglistBargan and DealOrNoDeal [31, 59]. Whilte these tasks do provide reward functions, they lack clarity on how to properly evaluate these rewards in an efficient and scalable manner without relying human evaluation, which is slow and costly. There have also been work on text-based games [19, 23, 63, 71] which do provide something similar to our datasets with sequential decision making tasks that can be efficiently evaluated. However, these tasks often constrain the actions the agents can take to a set of pre-selected options. Our benchmark differs from these in that we have carefully designed our LM task such that they are goal-directed, requires multi-step reasoning, and have well-defined metrics of success that can be easily evaluated. Our tasks also operate on unconstrained language inputs and outputs. Motivated by recent successes in using LLMs to generate synthetic data for RL [4, 19, 43], our dataset is generated using mostly synthetic data, and we provide fine-tuned LLM models to serve as the environment for efficient evaluation, which is imperative for RL which requires many environment interactions to properly learn.

RL for Language Models. RL for language models has received considerable attention in recent years due to RLHF (reinforcement learning with human feedback) [12], which is often used to align LLMs with human preferences [3, 4, 36, 41, 55, 67, 73]. However, these works do not focus on goal-directed tasks, as RLHF by design is a way to align a model with human preferences rather than a way to optimize for some given reward function for a task. They use RLHF as a way to fine-tune LLMs such that their outputs are more in line with human ethics. On the goal-directed end, RL in the form of bandit learning has been leveraged in several languag tasks with LLM, including machine translation [29, 38, 68], generation [34, 42, 45, 57], dialogue [14, 16, 27, 33, 55, 72], question answering [45], and summarization [6, 44, 69]. These include RL methods that learn by directly interacting with the environment (online RL) [10] and RL methods that only use a static dataset (offline RL) [1, 26, 27, 53, 59]. However, these tasks are single-step. Our benchmarks include multi-step tasks with well-defined reward functions, and each task has a simulator provided either by a language model or a scripted environment such that the evaluation-time behavior matches the system that produced the original training set. This makes it possible to rapidly iterate on algorithms with ground-truth evaluation. While ultimately RL methods might train LLMs on real-world datasets, such systems would likely need to be evaluated with user studies. We hope that our simulated benchmarks would serve as a high-fidelity and much more accessible mechanism to develop effective methods before committing to such costly evaluation protocols.

Capabilities of LLMs. There has been a surge in the capabilities of LLMs for generation [17, 24, 32, 46, 70], dialogue [20, 28, 31, 50, 51, 54], question answering [45], summarization [6, 44, 69], text-based games [19, 37], and translation [18]. However, these are often supervised-learning tasks that do not test the LLMs' abilities to achieve a specific long-term objective. Specifically, works in dialogue [20, 28, 50, 51] focuses on generating feasible-looking agent dialogue without explicit consideration for some multi-step objective, with the LLMs generating texts within the distribution of the training data [3, 55, 67, 73]. Our benchmarks allow for the development of algorithms that allow LLMs to *interact* in an environment to achieve some long-term objective.

3 Background

Preliminaries. We follow the standard RL setting: let $\mathcal{M} = (\mathcal{S}, \mathcal{A}, P, r, \rho, \gamma)$ denote a discounted MDP, which consists of a state space $s \in \mathcal{S}$, an action space $a \in \mathcal{A}$, a transition distribution $P(s'|s, a)$, a reward function $r(s, a)$, an initial state distribution $\rho(s_0)$, and a discount factor $\gamma \in (0, 1]$.

The goal of RL is to produce a policy π^* such that it maximizes the expected discounted sum of rewards over trajectories (τ) under the policy:

$$\pi^* = \arg\max_{\pi} \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{T-1} \gamma^t r_t\right], \text{where } \tau = (s_0, a_0, r_0, s_1, ..., a_{T-1}, r_{T-1}, s_T) \tag{.1}$$

RL for Multi-Turn Text Generation In a multi-turn text generation setting, states and actions consist of lists of text tokens. Let $\mathcal{V}$ denote the token vocabulary of the language models, then $\mathcal{S}, \mathcal{A} \subseteq \mathcal{V}^n$ where n denote the maximum number of tokens in a trajectory. Often, in dialogue settings, at every step, action tokens and new state tokens are appended to the previous state tokens to create the new state. This often result in the following state and action representations:

$$s_t = s_0^1 \dots s_0^{L_0} a_0^1 \dots a_0^{K_0} \dots s_{t-1}^1 \dots s_{t-1}^{L_{t-1}} a_{t-1}^1 \dots a_{t-1}^{K_{t-1}} s_t^1 \dots s_t^{L_t} \tag{.2}$$
$$a_t = a_t^1 \dots a_t^{K_t} \tag{.3}$$

where s_i^j and a_i^j denote the j-th token of the state or action at time step i, and L_i and K_i denote the length of the state or action at time step i.

4 Tasks and Datasets

In order to facilitate research in the goal-directed LLM RL setting, tasks and their associated datasets need to exhibit certain properties. We discuss these properties below.

Complex language. One clear distinguishing factor between a traditional RL setting and a goal-directed LLM RL setting is the use of language for states and actions. To fully leverage the capabilities of LLMs, states and actions, rather than selecting from a few sentence options, should be able to be any sentence without restrictions of length or content, or at least be varied enough such that the options are impossible to fully list out.

Complex decision making. RL shines in goal-directed tasks that require multi-step thinking and complex decision making. As discussed previously, prior works mostly focus on single-step tasks such as translation or non-goal-directed tasks such as chatting. An LLM RL task need to be goal-directed and take multiple steps to complete, where at each step the LLM need to not just imitate training data or human preference, but actually make decisions to achieve the final goal.

Easy to evaluate. A key component of training RL models is to evaluate, during and after training, how well your trained policy actually achieved its goals. As opposed to supervised learning, in RL, training/evaluation loss does not indicate policy performance [56]. Instead, the policy needs to interact with the environment to perform evaluation. This poses a problem since the environment for LLM RL tasks use language, and it is infeasible to have a human perform thousands of evaluation rollouts during and after training. Our tasks all include methods of simulating the environment with another LLM or carefully scripted environments to circumvent this issue. Although simulation does not match human natural language, it provides a strong enough indicator to benchmark whether an RL method is working [43].

Sparse rewards. Since dialogue is such a non-linear activity, you often do not know how well your conversation achieved your goal until the end of the conversation. This is also true for many non-dialogue tasks as well. For this reason, tasks and algorithms to solve these tasks need to perform in the sparse rewards case where a reward signal is only given at the end of a trajectory/conversation.

Suboptimal data. For an offline RL dataset to truly test the capabilities of RL, the dataset need to include suboptimal data [15]. This also makes the dataset more varied and easier to collect.

The same is true for the LLM RL setting. Our datasets contain trajectories of a wide range of performances.

4.1 Twenty Questions

Twenty questions is a "rational human" task. This task specifically tests "complex decision making" to see if a policy can successfully reason about an unknown subject based on context to determine what it is.

Task Description. This task simulates twenty questions, a classic guessing game in which one player (the oracle) thinks of an object, and the other player (the guesser) tries to guess what it is by asking a series of yes-or-no questions. The game continues with the guesser asking a series of questions until they either guess the correct answer or run out of questions. The oracle will respond truthfully to each question. The goal is to train a guesser model that will ask questions to determine what the object is within twenty questions using as few questions as possible.

The dataset consists of 100K full conversations between the guesser and the oracle. The oracle can choose from a set of 158 unique objects taken from 17 different category of objects/animals. See Appendix 1.1 for the full list of objects. Each object have a roughly equal amount of conversations in the dataset, but varies in terms of how many conversations are successful in guessing the object. However, every object has at least one conversation where it is guessed correctly to facilitate learning.

Data Collection. To collect the data, we used LLMs. Since OpenAI's GPT-3.5 generated reasonable questions and answers when used out-of-the-box, we used it to collect a initial dataset. We collected 1K conversations by querying GPT-3.5 (specifically `text-davinci-003`) [8], alternating between the guesser and oracle. The guesser used a temperature of 0.7 to create varied questions, whereas the oracle used a temperature of 0 to provide a ground truth answer to each question. It was important to separate the guesser and oracle since we did not want the guesser to know the object in the oracle's prompt. Using these conversations after some filtering, we fine-tuned a FLAN-T5-XL guesser model and a FLAN-T5-XL oracle model. Using these models, we generated the final 100K conversations dataset by having the two models talk to each other.

See Appendix 1.2 for the exact prompts used for both the guesser and oracle to generate the data. See Appendix 1.3 for example conversations.

Reward Function. For the reward function, since we want the guesser to guess the correct word in as few guesses as possible, the reward function reflects this by penalizing the guesser for each question that does not guess the correct word.

$$r(\texttt{question}) = \begin{cases} 0 & \text{if } \texttt{question} \text{ correctly guessed the word} \\ -1 & \text{otherwise} \end{cases} \tag{.4}$$

If the guesser model correctly guessed the word, then the trajectory ends.

Over twenty questions, the maximum total sum of rewards is 0 if the guesser guessed the word on the first question, whereas the minimum is −20 if the guesser did not guess the word in twenty questions.

Dataset statistics. See Table 1 for the statistics of the dataset.

# convs	100000
# objects/animals	158
avg returns	−17.39
std returns	4.56
success rate	0.3097

Table 1: Statistics of the twenty questions dataset.

4.2 Car Dealer

Car dealer is a "irrational human" task. In this task the agents don't necessarily make rational decisions or say sentences in a structure manner. This task requires more complex language parsing and generation.

Task Description. This task simulates a conversation between a car buyer with different personalities and a car seller. More specifically, the buyer wants to buy a certain type of car within a certain budget, and the seller wants to complete the purchase ideally with a high sale price. The goal is to train a seller model that can accommodate buyers with different personalities and complete the purchase.

The dataset contains 19408 conversations of 10 different buyer personalities: {`abusive`, `angry`, `insulting`, `polite`, `respectful`, `rude`, `sarcastic`, `talkative`, `toxic`, `uncommunicative`}.

Each conversation has one personality chosen for the buyer. We chose a wide variety of personalities in order to create more variations in the kind of conversations that occurs in the dataset. We note that most of these personalities are very extreme, because we found empirically that extreme personalities tend to create a much larger variance in the kind of conversations generated by ChatGPT. See Appendix 2.1 for example conversations with each personality type. The dataset also contains other characteristics of the conversations such as desired brands, features, classifications (i.e. car or truck), and budgets, as follows:

- Brands: BMW, Lexus, Honda, Toyota, Mazda, Audi, Hyundai, Porsche, Tesla, Volkswagen, Ford, Mercedes-Benz, Subaru, Porsche.
- Features: leather seats, sunroof, heated seats, backup camera, navigation system, blind spot monitoring, third-row seating, Apple CarPlay.
- Classifications: SUV, sedan, truck, convertible, luxury, electric.
- Budgets: $10000, $30000, $50000, $70000, $90000.

The brands were chosen in such a way to cover a wide range of prices, from the more economic brands such as Honda and Toyota, to the more luxurious brands such as Porsche and Tesla. The list of budgets were chosen to cover the range of prices that our chosen brands usually cover. We found that the choice of brand and budget significantly impacted the rewards (for both reward functions we propose below) and the MSRP of the final suggested cars (see Table 4 Table 5), but did not significantly affect the rate of whether the car was bought or not (see Table 6).

Since our goal is to train a persuasive seller model that can accommodate different personalities, each conversation also contains the following output: (1) `car_bought`, a binary variable indicating whether the purchase is successful in one conversation; (2) `msrp`, a number that represents the MSRP (manufacturer suggested retail price) of the car of interest; (3) `buy_price`, a number that represents the sale price if the deal is successful.

Data Collection. We collect our data using ChatGPT [39] (specifically `gpt-3.5-turbo`). For each conversation, we randomly sample one desired `brand`, one desired `classification`, one `budget`, and 1-4 desired `features`. Each conversation was then generated in a single shot with the following prompt using temperature of 0.8, filling in variables as denoted with `{{variable}}`.

```
Generate a conversation between an agent at a car dealership and a buyer.
The buyer prefers {{brand}} {{classification}} with the following features:
{{features}}. However, the buyer could be convinced to purchase something
```

```
else. The buyer has a budget of ${{budget}}:, but the agent does not know the
budget beforehand. The agent can try to convince the buyer to go over their
budget, but the buyer could also end the conversation if pushed too hard.
The agent wants to sell the car but also will refuse to sell the car too far
below the MSRP. The agent is also trying to maximize the sell price of the
car. The buyer's personality is extremely {{personality}}, so the agent would
have to navigate this personality in order to sell the car. The conversation
can only end if the buyer agrees to buy the car or decides to not buy the car.

Do not include any off-screen or non-dialogue interactions. Start every
line with "Agent:" or "Buyer:", alternating between agent and buyer.

At the end of the conversation, output whether the buyer decided to buy
the car or not and the MSRP of the car of interest. If they bought the car,
also output the price the buyer bought the car for, in the following format:
"Car Bought: Yes/No, MSRP: $price, Buy Price: $price".
```

For each of the 10 personalities, we collected 2000 conversations in such a manner. Since we explicitly asked ChatGPT to output the evaluation metrics (`car_bought`, `msrp`, `buy_price`) at the end of each conversation, we can extract these metrics after generating each conversation. After collecting all 20K conversations, we further cleaned up the data by removing the conversations without proper metrics or containing other errors, resulting in the dataset with a size of 19408.

Reward Functions Since the purpose of this task is to train a persuasive seller model that can be invariant to the personality of the dealer, we propose two sparse reward functions: (1) r_{per} and (2) r_{rev}.

(1) r_{per} is designed based on the two following intuitions:

- When the deal is successful, the seller should be awarded if it persuades the buyer to go beyond their budget or it sells the car beyond the MSRP. i.e., the reward should increase if the ratios `buy_price/budget` or `buy_price/msrp` increases.
- When the deal is not successful, the seller should be penalized more if the buyer has a higher budget than the MSRP. i.e., the reward should decrease when the ratio `budget/msrp` increases.

Based on this, we propose the following sparse reward functions, where the agent only observes the reward at the end of each conversation:

$$r_{\mathtt{per}}(\mathtt{car\_bought}, \mathtt{buy\_price}, \mathtt{msrp}, \mathtt{budget}) = \begin{cases} \frac{\mathtt{buy\_price}}{0.5(\mathtt{msrp}+\mathtt{budget})} & \text{if } \mathtt{car\_bought} = \mathtt{True} \\ -\frac{\mathtt{budget}-\mathtt{msrp}}{\mathtt{msrp}} & \text{otherwise.} \end{cases} \quad (.5)$$

(2) $r_{\mathtt{rev}}$ is much simpler in that it only cares about revenue without regards to the buyer's circumstances. It is much simpler but has much higher variance since the final buy price of the car is correlated with the buyer's budget. This is also a sparse reward function where the agent only observes the reward at the end of each conversation.

$$r_{\mathtt{rev}}(\mathtt{car\_bought}, \mathtt{buy\_price}, \mathtt{msrp}, \mathtt{budget}) = \begin{cases} \mathtt{buy\_price} & \text{if } \mathtt{car\_bought} = \mathtt{True} \\ 0 & \text{otherwise.} \end{cases} \quad (.6)$$

Dataset Statistics. We present some statistics of the Car Dealer dataset. Table 2 shows that the valid conversations of different personalities are roughly at the same scales, which demonstrates the viability of our prompt. Interestingly, Table 2 also suggests that the `success rate`, largely varies between different personalities, making this a varied task.

	all	`abu.`	`ang.`	`ins.`	`pol.`	`res.`	`rud.`	`sar.`	`tal.`	`tox.`	`unc.`
# convs	19408	1955	1954	1926	1937	1951	1950	1932	1943	1942	1918
sell rate	0.524	0.459	0.559	0.382	0.592	0.577	0.362	0.808	0.724	0.350	0.424
avg $r_{\mathtt{per}}$	0.524	0.445	0.525	0.367	0.550	0.523	0.362	0.767	0.693	0.358	0.385
std $r_{\mathtt{per}}$	0.586	0.590	0.583	0.610	0.556	0.571	0.588	0.485	0.516	0.575	0.612
avg $r_{\mathtt{rev}}$	24309	21676	26021	18183	26449	25619	17244	38312	33932	16720	18920
std $r_{\mathtt{rev}}$	29016	28821	29443	27771	28663	28107	27160	28238	29093	27052	27320

Table 2: Statistics of the car dealer task of different personalities. The abbreviations in the first row represent our personalities {`abusive, angry, insulting, polite, respectful, rude, sarcastic, talkative, toxic, uncommunicative`}.

Our data also shows that ChatGPT intrinsically know the real MSRP of different cars. The average `msrp` of different brands in Table 3 demonstrates that luxurious brands such as `Porsche` and `Tesla` are more expensive than classic high-end brands (`Audi`, `BMW`, `Mercedes-Benz`, and `Lexus`). The `msrp` of High-end brands are also more expensive than the remaining economic brands. Table 5 shows the average MSRP of each brand at different budget levels, which also shows that brands such as `Tesla` doesn't have lower budget options.

Brands	`Aud.`	`BMW`	`For.`	`Hon.`	`Hyu.`	`Lex.`	`Maz.`
Avg. `msrp`	50703	51550	46239	41416	41116	49247	41753
Brands cont.	`Mer.`	`Por.`	`Sub.`	`Tes.`	`Toy.`	`Vol.`	
Avg. `msrp`	51984	61215	39954	62175	41441	43111	

Table 3: Average `msrp` of different car brands. The abbreviations in the first row represent the brands {`Audi, BMW, Honda, Hyundai, Lexus, Mazda, Mercedes-Benz, Porsche, Subaru, Tesla, Toyota, Volkswagen`}.

	r_{per}					r_{rev}				
budget	\$10000	\$30000	\$50000	\$70000	\$90000	\$10000	\$30000	\$50000	\$70000	\$90000
Audi	0.705	0.566	0.540	0.459	0.451	7482	16052	26744	33149	44215
BMW	0.740	0.609	0.548	0.466	0.418	7609	17752	27312	34041	42532
Ford	0.613	0.599	0.452	0.381	0.364	5517	17482	24288	32780	43793
Honda	0.598	0.543	0.391	0.176	0.048	5006	15864	23565	26945	32957
Hyundai	0.685	0.529	0.385	0.374	0.144	6524	15622	22820	34140	36548
Lexus	0.724	0.593	0.535	0.468	0.364	6771	16865	27021	35768	40292
Mazda	0.652	0.513	0.353	0.349	0.172	6206	15018	21788	32678	37318
Mercedes	0.732	0.638	0.567	0.495	0.519	7917	18671	27743	35746	48274
Porsche	0.849	0.605	0.578	0.586	0.473	11293	16146	26538	39991	43162
Subaru	0.634	0.547	0.411	0.283	0.017	5662	16266	23649	31260	30394
Tesla	0.920	0.619	0.535	0.503	0.444	13120	14241	25884	35881	43445
Toyota	0.645	0.599	0.381	0.295	0.143	5690	17509	21893	28979	36196
Volkswagen	0.648	0.581	0.379	0.353	0.115	6102	17104	22464	31588	32086

Table 4: Average rewards (for both r_{per} and r_{rev}) for each buyer preferred brand of car at a given buyer budget level.

budget	$10000	$30000	$50000	$70000	$90000
Audi	25552	36488	49922	65552	76678
BMW	26721	36005	51516	64650	79718
Ford	17084	32244	46527	60364	71693
Honda	17920	31450	43022	53850	61736
Hyundai	17866	30569	43251	54360	62437
Lexus	24546	35891	49952	62490	76488
Mazda	17335	31187	42889	54790	63026
Mercedes	29494	35092	51958	67286	79523
Porsche	44258	42437	60082	74038	88319
Subaru	17171	31096	42605	52056	56808
Tesla	52568	46439	56057	70942	83442
Toyota	18775	31698	42377	53486	62349
Volkswagen	17470	31312	43396	55531	66666

Table 5: Average MSRP for each buyer preferred brand of car at a given buyer budget level.

budget	$10000	$30000	$50000	$70000	$90000
Audi	0.442	0.507	0.565	0.526	0.578
BMW	0.486	0.570	0.557	0.537	0.539
Ford	0.449	0.585	0.520	0.541	0.595
Honda	0.404	0.540	0.540	0.485	0.520
Hyundai	0.507	0.544	0.529	0.604	0.567
Lexus	0.453	0.536	0.569	0.570	0.524
Mazda	0.498	0.515	0.511	0.587	0.580
Mercedes	0.423	0.597	0.572	0.547	0.616
Porsche	0.436	0.495	0.513	0.578	0.504
Subaru	0.456	0.551	0.557	0.579	0.528
Tesla	0.355	0.427	0.504	0.527	0.519
Toyota	0.461	0.597	0.527	0.549	0.548
Volkswagen	0.493	0.581	0.515	0.561	0.473

Table 6: Average success rate for whether the seller sold a car or not for each buyer preferred brand of car at a given buyer budget level.

5 Benchmarking Prior Methods

We evaluate each of our tasks and dataset with a combination of online and offline RL algorithms: behavioral cloning (BC), top-10% behavioral cloning (BC-10%), Monte-Carlo returns (MC), Implicit Language Q-Learning (ILQL) [53], and Proximal Policy Optimization (PPO) [49]. We use the decoder-only transformer model GPT2 [46] (and its variant sizes) as the policy and value network for our experiments. We give a brief explanation the RL algorithms and how we apply them to the RL setting below.

BC. In line with standard RL nomenclature, we denote supervised finetuning as behavioral cloning (BC). We use a decoder-only transformer model, in this case GPT2, as our policy. The reason we use a decoder-only model rather than an encoder-decoder model is for more efficient training. Naïvely, an encoder-decoder model naturally models an RL policy $a_t \sim \pi(\cdot|s_t)$, where the state text tokens s_t can be input into the encoder, and the decoder will output the action text tokens a_t. However, as we described in the Section 3 (Background), when doing RL for multi-turn text generation, s_t is the concatenation of all previous state and action text tokens. This means that we can instead leverage the causal-attention masking of the GPT2 architecture. Instead of using (s_t, a_t) for all $t = 0, ..., T-1$ as the input output pair for training, we can simply use (s_{T-1}, a_{T-1}) as the single input output pair for an entire trajectory. When performing rollouts with the learned policy, we can simply use the autoregressive nature of the GPT2 model to generate the a_t given s_t. A note for generation is that there must be a unique end-of-sentence token appended to the end of every a_t such that the policy knows when to stop generating.

For BC training, unlike standard text generation, loss is only computed on action output tokens, the state outputs are masked out. The model uses a standard language model (LM) head (i.e. one logits output for each token in the vocabularly), and the loss for BC is the standard cross-entropy loss since BC is just supervised learning. For training, we also initialize the policy using the public pretrained GPT2 weights [46].

We chose BC as a baseline because this is the most basic baseline. Since we collected data using LLMs, we should expect that a LLM policy should be able to immidate the training data behavior.

BC-10%. BC-10% is supervised fine-tuning on filtered data. The method is the same as BC, but the model is only trained on the top 10% of the dataset in terms of total returns. We chose BC-10% as a baseline because it is another common offline RL baseline, as it is similar to training on expert data assuming the top 10% of the dataset is expert-level.

MC. For Monte-Carlo returns (MC), we initialize the GPT2 model with weights from a trained BC model. We then freeze the model weights and train two state-value function (Q) heads and one value function (V) head (in addition to the standard LM head). The targets for these head is return-to-go:

$$R_t = \sum_{i=t}^{T-1} \gamma^{i-t} r_t \tag{.7}$$

and we use MSE loss for both Q and V:

$$J(Q) = \mathbb{E}_{(s_t, a_t, r_{t:T-1}) \sim \mathcal{D}} \left[(Q(s_t, a_t) - R_t)^2 \right] \tag{.8}$$

$$J(V) = \mathbb{E}_{(s_t, r_{t:T-1}) \sim \mathcal{D}} \left[(V(s_t) - R_t)^2 \right] \tag{.9}$$

where $\mathcal{D}$ represents the dataset. In MC, $Q(s_t, a_t)$ represents how much more rewards the policy will get if it takes action a_t at the state s_t under some policy (in this case the policy that collected the dataset). $V(s_t)$ represents how much more rewards the policy will get from this current state under that same policy,

We use the same training procedure as BC, which means that during training, Q and V for every time-step of a trajectory is calculated simultaneously in one forward pass, with their loss taken simultaneously as well.

During rollout, when sampling, we perturb the base BC policy with the learned value-functions [53]. Let π_β represent the policy trained with BC, and α represent a scalar multiplier, then:

$$\pi_{\mathsf{MC}}(a_t|s_t) \propto \pi_\beta(a_t|s_t)^{\alpha(Q(s_t,a_t)-V(s_t))} \tag{.10}$$

We chose MC as a baseline because it is a rudimentary value-based RL method to demonstrate the baseline capabilities of RL.

ILQL. Implicit Language Q-Learning (ILQL) [53] is an offline RL algorithm specifically for language models. The setup for training and rollout is the same as MC, with two Q-functions, one V-function, and perturbation based policy extraction as Equation .10. However there are a few key differences. (1) The value functions share a separate transformer network from the base policy model. (2) The loss function for Q and V is different, and uses a "target network" for bootstrapping, which means that the loss target includes itself, which could lead to instability. (3) there is an additional entropy regularization loss term taken from CQL [30]. The exact details of the ILQL algorithm is out of the scope of this report. Please refer to the ILQL paper for full details [53].

We chose ILQL as a baseline because it is a method specifically made for training language-based RL tasks with LLMs. It has also shown impressive results.

PPO . Proximal Policy Optimization (PPO) [49] is an online RL algorithm. This means that unlike the previous four methods which uses a fixed dataset, PPO needs to perform environment rollouts during training to collect training data. The policy setup is the same as BC, where we use a decoder-only transformer network initialized with pretrained GPT2 weights, but there is also a value function head (known as the terminal value function). The value function head is setup the same as MC, where it outputs a value at every time step. The value function loss uses a variant of return-to-go called generalized advantage estimate (GAE). The exact details of the PPO algorithm is out of the scope of this report. Please refer to the PPO paper for full details [49].

We chose PPO as a baseline because it is one of the most widely used and stable online RL algorithms. LLM training are often unstable, so PPO should alleviate a lot of the instability. PPO also does not use boostrapping (i.e. using a value function who in its loss term contains itself), which should make it more stable as well.

5.1 Twenty Questions

For twenty questions, we used GPT2-medium as the guesser model. We chose GPT2-medium because the task's language component was simple, as the sentences have a somewhat clear structure. For training, we swept over multiple hyper-parameters for each algorithm and report the result for the best one from each algorithm. See Table 9 for the hyperparameters breakdown.

For evaluation, we used greedy sampling for the policy. Since the oracle is deterministic and the policy (when using greedy sampling) is also deterministic, we only needed to evaluate each word once to get the average returns for each policy. The results are shown in Table 7.

BC	BC-10%	MC	ILQL	PPO
-16.0	-14.6	-13.9	-14.2	-14.9

Table 7: Average returns for twenty questions achieved by the baseline algorithms

5.2 Car Dealer

For car dealer, we used GPT2-XL as the seller model. We chose GPT2-XL because the task was much more complex in its language component, being both longer and less structured in its conversations. We trained each algorithm for both reward functions we proposed. For training, we swept over multiple hyper-parameters for each algorithm and report the result for the best one from each algorithm. See Table 9 for the hyperparameters breakdown.

For evaluation, we took the dataset and trained a GPT2-XL buyer model using BC. We then evaluated all the seller models against this BC buyer. The training for the BC seller and buyer is symmetrical. It is also reasonable (and perhaps better) to train the buyer model using some other algorithm that is less exploitable. However, for the purpose of this benchmark, we simply provide one way of doing evaluation. An important detail is that the buyer model can at any point can output a "stop" token and then output the values needed for calculating reward, i.e. `car_bought`, `buy_price`, and `msrp`.

We sampled 4000 trajectories for each policy and averaged the final reward for each. The results are shown in Table 8.

	BC	BC-10%	MC	ILQL	PPO
r_{per}	0.416	0.785	0.446	0.463	0.549
r_{rev}	21940	75708	23984	27399	27391

Table 8: Average reward for car dealer achieved by the baseline algorithms

We note here that BC-10% is especially able to exploit our choice of the buyer model, whereas the other methods could not. This is an interesting research direction to create RL algorithms that can create LLM policies that can exploit other LLM policies. Another direction is to pick an algorithm to train the buyer model to resist this type of exploitation.

		20 Questions	Car Dealer r_{per}	Car Dealer r_{rev}
BC	model	gpt2-medium	gpt2-xl	gpt2-xl
	lr	1e-4	1e-4	1e-4
	batch size	128	128	128
BC-10%	model	gpt2-medium	gpt2-xl	gpt2-xl
	lr	1e-4	1e-4	1e-4
	batch size	128	128	128
MC	model	gpt2-medium	gpt2-xl	gpt2-xl
	lr	1e-4	1e-4	1e-4
	batch size	128	128	128
	α	16	16	4
	discount γ	0.99	0.99	0.99
ILQL	model	gpt2-medium	gpt2-xl	gpt2-xl
	lr	1e-4	1e-4	1e-4
	batch size	128	128	128
	α	4	8	1
	cql weight	0.001	0.001	0.001
	expectile	0.7	0.7	0.7
	discount γ	0.99	0.99	0.99
PPO	model	gpt2-medium	gpt2-xl	gpt2-xl
	lr	1e-6	5e-6	5e-6
	batch size	2048	4000	4000
	steps per batch	1	4	4
	GAE λ	0.95	0.95	0.95
	discount γ	0.99	0.99	0.99
	KL coef.	0.01	0.01	0.01
	clip range	0.2	0.2	0.2

Table 9: Hyperparameters for baseline experiments.

6 Discussions

We have proposed a set of benchmarking tasks and datasets for the goal-directed LLM RL domain. The tasks are chosen in such a way to cover a broad set of properties we believe make a good LLM RL task. For each of the tasks, we describe in detail the data collection procedure using existing LLMs in the hopes that future research can expand on these tasks easier.

RL for goal-directed language tasks using LLMs is a fascinating new field that has not seen a lot of progress, and we hope that this benchmark can accelerate it to new grounds.

This work is also part of a much larger benchmark we are actively working on that includes many more tasks, along with a code base which integrates these datasets and allows easy training of LLMs for RL.

Future works include:

- Creating new tasks to test more aspects of the domain.
- Providing better ways to design simulators for dialogue tasks to prevent exploitation issues.
- Using this benchmark to design better algorithms to train goal-directed LLM RL models.

Bibliography

1. M. F. A. R. D. T. (FAIR)†, A. Bakhtin, N. Brown, E. Dinan, G. Farina, C. Flaherty, D. Fried, A. Goff, J. Gray, H. Hu, A. P. Jacob, M. Komeili, K. Konath, M. Kwon, A. Lerer, M. Lewis, A. H. Miller, S. Mitts, A. Renduchintala, S. Roller, D. Rowe, W. Shi, J. Spisak, A. Wei, D. Wu, H. Zhang, and M. Zijlstra. "Human-level play in the game of <i>Diplomacy</i> by combining language models with strategic reasoning". In: *Science* 378:6624, 2022, pp. 1067–1074. DOI: 10.1126/science.ade9097. eprint: https://www.science.org/doi/pdf/10.1126/science.ade9097. URL: https://www.science.org/doi/abs/10.1126/science.ade9097.

2. O. M. Andrychowicz, B. Baker, M. Chociej, R. Jozefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, et al. "Learning dexterous in-hand manipulation". In: *The International Journal of Robotics Research* 39:1, 2020, pp. 3–20.

3. Y. Bai, A. Jones, K. Ndousse, A. Askell, A. Chen, N. DasSarma, D. Drain, S. Fort, D. Ganguli, T. Henighan, N. Joseph, S. Kadavath, J. Kernion, T. Conerly, S. El-Showk, N. Elhage, Z. Hatfield-Dodds, D. Hernandez, T. Hume, S. Johnston, S. Kravec, L. Lovitt, N. Nanda, C. Olsson, D. Amodei, T. Brown, J. Clark, S. McCandlish, C. Olah, B. Mann, and J. Kaplan. *Training a Helpful and Harmless Assistant with Reinforcement Learning from Human Feedback*. 2022. arXiv: 2204.05862 [cs.CL].

4. Y. Bai, S. Kadavath, S. Kundu, A. Askell, J. Kernion, A. Jones, A. Chen, A. Goldie, A. Mirhoseini, C. McKinnon, C. Chen, C. Olsson, C. Olah, D. Hernandez, D. Drain, D. Ganguli, D. Li, E. Tran-Johnson, E. Perez, J. Kerr, J. Mueller, J. Ladish, J. Landau, K. Ndousse, K. Lukosuite, L. Lovitt, M. Sellitto, N. Elhage, N. Schiefer, N. Mercado, N. DasSarma, R. Lasenby, R. Larson, S. Ringer, S. Johnston, S. Kravec, S. E. Showk, S. Fort, T. Lanham, T. Telleen-Lawton, T. Conerly, T. Henighan, T. Hume, S. R. Bowman, Z. Hatfield-Dodds, B. Mann, D. Amodei, N. Joseph, S. McCandlish, T. Brown, and J. Kaplan. *Constitutional AI: Harmlessness from AI Feedback*. 2022. arXiv: 2212.08073 [cs.CL].

5. C. Berner, G. Brockman, B. Chan, V. Cheung, P. Dębiak, C. Dennison, D. Farhi, Q. Fischer, S. Hashme, C. Hesse, et al. "Dota 2 with large scale deep reinforcement learning". In: *arXiv preprint arXiv:1912.06680*, 2019.

6. F. Böhm, Y. Gao, C. M. Meyer, O. Shapira, I. Dagan, and I. 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)*. Association for Computational Linguistics, Hong Kong, China, 2019, pp. 3110–3120. DOI: 10.18653/v1/D19-1307. URL: https://aclanthology.org/D19-1307.

7. O. r. Bojar, R. Chatterjee, C. Federmann, Y. Graham, B. Haddow, M. Huck, A. Jimeno Yepes, P. Koehn, V. Logacheva, C. Monz, M. Negri, A. Neveol, M. Neves, M. Popel, M. Post, R. Rubino, C. Scarton, L. Specia, M. Turchi, K. Verspoor, and M. Zampieri. "Findings of the 2016 Conference on Machine Translation". In: *Proceedings of the First Conference on Machine Translation*. Association for Computational Linguistics, Berlin, Germany, 2016, pp. 131–198. URL: http://www.aclweb.org/anthology/W/W16/W16-2301.

8. T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei. *Language Models are Few-Shot Learners*. 2020. arXiv: 2005.14165 [cs.CL].

9. CarperAI. *Transformer Reinforcement Learning X*. 2022. URL: https://github.com/CarperAI/trlx.

10. T. Carta, C. Romac, T. Wolf, S. Lamprier, O. Sigaud, and P.-Y. Oudeyer. *Grounding Large Language Models in Interactive Environments with Online Reinforcement Learning*. 2023. arXiv: 2302.02662 [cs.LG].

11. M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. de Oliveira Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, A. Ray, R. Puri, G. Krueger, M. Petrov, H. Khlaaf, G. Sastry, P. Mishkin, B. Chan, S. Gray, N. Ryder, M. Pavlov, A. Power, L. Kaiser, M. Bavarian, C. Winter, P. Tillet, F. P. Such, D. Cummings, M. Plappert, F. Chantzis, E. Barnes, A. Herbert-Voss, W. H. Guss, A. Nichol, A. Paino, N. Tezak, J. Tang, I. Babuschkin, S. Balaji, S. Jain, W. Saunders, C. Hesse, A. N. Carr, J. Leike, J. Achiam, V. Misra, E. Morikawa, A. Radford, M. Knight, M. Brundage, M. Murati, K. Mayer, P. Welinder, B. McGrew, D. Amodei, S. McCandlish, I. Sutskever, and W. Zaremba. *Evaluating Large Language Models Trained on Code*. 2021. arXiv: 2107.03374 [cs.LG].

12. P. Christiano, J. Leike, T. B. Brown, M. Martic, S. Legg, and D. Amodei. *Deep reinforcement learning from human preferences*. 2023. arXiv: 1706.03741 [stat.ML].

13. K. Cobbe, V. Kosaraju, M. Bavarian, M. Chen, H. Jun, L. Kaiser, M. Plappert, J. Tworek, J. Hilton, R. Nakano, et al. "Training verifiers to solve math word problems". In: *arXiv preprint arXiv:2110.14168*, 2021.

14. H. Cuayáhuitl, S. Keizer, and O. Lemon. *Strategic Dialogue Management via Deep Reinforcement Learning*. 2015. arXiv: 1511.08099 [cs.AI].

15. J. Fu, A. Kumar, O. Nachum, G. Tucker, and S. Levine. *D4RL: Datasets for Deep Data-Driven Reinforcement Learning*. 2021. arXiv: 2004.07219 [cs.LG].

16. K. Georgila and D. Traum. "Reinforcement Learning of Argumentation Dialogue Policies in Negotiation." In: 2011, pp. 2073–2076. DOI: 10.21437/Interspeech.2011-544.

17. M. Ghazvininejad, X. Shi, J. Priyadarshi, and K. Knight. "Hafez: an Interactive Poetry Generation System". In: *Proceedings of ACL 2017, System Demonstrations*. Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 43–48. URL: https://aclanthology.org/P17-4008.

18. J. Gu, K. Cho, and V. O. Li. "Trainable Greedy Decoding for Neural Machine Translation". In: *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Copenhagen, Denmark, 2017, pp. 1968–1978. DOI: 10.18653/v1/D17-1210. URL: https://aclanthology.org/D17-1210.

19. M. Hausknecht, P. Ammanabrolu, C. Marc-Alexandre, and Y. Xingdi. "Interactive Fiction Games: A Colossal Adventure". In: *CoRR* abs/1909.05398, 2019. URL: http://arxiv.org/abs/1909.05398.

20. H. He, D. Chen, A. Balakrishnan, and P. Liang. *Decoupling Strategy and Generation in Negotiation Dialogues*. 2018. arXiv: 1808.09637 [cs.CL].

21. D. Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt. "Measuring massive multitask language understanding". In: *arXiv preprint arXiv:2009.03300*, 2020.

22. D. Hendrycks, C. Burns, S. Kadavath, A. Arora, S. Basart, E. Tang, D. Song, and J. Steinhardt. "Measuring mathematical problem solving with the math dataset". In: *arXiv preprint arXiv:2103.03874*, 2021.

23. D. Hendrycks, M. Mazeika, A. Zou, S. Patel, C. Zhu, J. Navarro, D. Song, B. Li, and J. Steinhardt. "What Would Jiminy Cricket Do? Towards Agents That Behave Morally". In: *NeurIPS*, 2021.

24. A. Holtzman, J. Buys, M. Forbes, A. Bosselut, D. Golub, and Y. Choi. "Learning to Write with Cooperative Discriminators". In: *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, Melbourne, Australia, 2018, pp. 1638–1649. DOI: 10.18653/v1/P18-1152. URL: https://aclanthology.org/P18-1152.

25. R. Irvine, D. Boubert, V. Raina, A. Liusie, V. Mudupalli, A. Korshuk, Z. Liu, F. Cremer, V. Assassi, C.-C. Beauchamp, et al. "Rewarding Chatbots for Real-World Engagement with Millions of Users". In: *arXiv preprint arXiv:2303.06135*, 2023.

26. Y. Jang, J. Lee, and K.-E. 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.

27. 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". In: *Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

28. N. Jaques, S. Gu, D. Bahdanau, J. M. Hernández-Lobato, R. E. Turner, and D. Eck. "Sequence Tutor: Conservative Fine-Tuning of Sequence Generation Models with KL-control". In: *Proceedings of the 34th International Conference on Machine Learning*. Ed. by D. Precup and Y. W. Teh. Vol. 70. Proceedings of Machine Learning Research. PMLR, 2017, pp. 1645–1654. URL: https://proceedings.mlr.press/v70/jaques17a.html.

29. S. Kiegeland and J. 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*. Association for Computational Linguistics, Online, 2021, pp. 1673–1681. DOI: 10.18653/v1/2021.naacl-main.133. URL: https://aclanthology.org/2021.naacl-main.133.

30. A. Kumar, A. Zhou, G. Tucker, and S. Levine. *Conservative Q-Learning for Offline Reinforcement Learning*. 2020. arXiv: 2006.04779 [cs.LG].

31. M. Lewis, D. Yarats, Y. N. Dauphin, D. Parikh, and D. Batra. *Deal or No Deal? End-to-End Learning for Negotiation Dialogues*. 2017. arXiv: 1706.05125 [cs.AI].

32. J. Li, W. Monroe, and D. Jurafsky. *Learning to Decode for Future Success*. 2017. arXiv: 1701.06549 [cs.CL].

33. J. Li, W. Monroe, A. Ritter, M. Galley, J. Gao, and D. Jurafsky. *Deep Reinforcement Learning for Dialogue Generation*. 2016. arXiv: 1606.01541 [cs.CL].

34. X. Lu, S. Welleck, J. Hessel, L. Jiang, L. Qin, P. West, P. Ammanabrolu, and Y. Choi. *Quark: Controllable Text Generation with Reinforced Unlearning*. 2022. arXiv: 2205.13636 [cs.CL].

35. S. Mishra, D. Khashabi, C. Baral, and H. Hajishirzi. "Cross-task generalization via natural language crowdsourcing instructions". In: *ACL*. 2022.

36. R. Nakano, J. Hilton, S. Balaji, J. Wu, L. Ouyang, C. Kim, C. Hesse, S. Jain, V. Kosaraju, W. Saunders, X. Jiang, K. Cobbe, T. Eloundou, G. Krueger, K. Button, M. Knight, B. Chess, and J. Schulman. *WebGPT: Browser-assisted question-answering with human feedback*. 2022. arXiv: 2112.09332 [cs.CL].

37. K. Narasimhan, T. Kulkarni, and R. Barzilay. "Language Understanding for Text-based Games using Deep Reinforcement Learning". In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Lisbon, Portugal, 2015, pp. 1–11. DOI: 10.18653/v1/D15-1001. URL: https://aclanthology.org/D15-1001.

38. K. Nguyen, H. Daumé III, and J. 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*. Association for Computational Linguistics, Copenhagen, Denmark, 2017, pp. 1464–1474. DOI: 10.18653/v1/D17-1153. URL: https://aclanthology.org/D17-1153.

39. OpenAI. *ChatGPT*. 2022. URL: https://openai.com/blog/chatgpt.

40. OpenAI. *GPT-4*. 2023. URL: https://openai.com/research/gpt-4.

41. L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, J. Schulman, J. Hilton, F. Kelton, L. Miller, M. Simens, A. Askell, P. Welinder, P. Christiano, J. Leike, and R. Lowe. *Training language models to follow instructions with human feedback*. 2022. arXiv: 2203.02155 [cs.CL].

42. R. Y. Pang and H. He. "Text Generation by Learning from Demonstrations". In: *International Conference on Learning Representations*. 2021. URL: https://openreview.net/forum?id=RovX-uQ1Hua.

43. J. S. Park, J. C. O'Brien, C. J. Cai, M. R. Morris, P. Liang, and M. S. Bernstein. *Generative Agents: Interactive Simulacra of Human Behavior*. 2023. arXiv: 2304.03442 [cs.HC].

44. R. Paulus, C. Xiong, and R. Socher. *A Deep Reinforced Model for Abstractive Summarization*. 2017. arXiv: 1705.04304 [cs.CL].

45. V. Pyatkin, J. D. Hwang, V. Srikumar, X. Lu, L. Jiang, Y. Choi, and C. Bhagavatula. *Reinforced Clarification Question Generation with Defeasibility Rewards for Disambiguating Social and Moral Situations.* 2022. arXiv: 2212.10409 [cs.CL].

46. A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al. "Language models are unsupervised multitask learners". In: *OpenAI blog* 1:8, 2019, p. 9.

47. P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. "Squad: 100,000+ questions for machine comprehension of text". In: *arXiv preprint arXiv:1606.05250*, 2016.

48. R. Ramamurthy, P. Ammanabrolu, K. Brantley, J. Hessel, R. Sifa, C. Bauckhage, H. Hajishirzi, and Y. 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.

49. J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. "Proximal policy optimization algorithms". In: *arXiv preprint arXiv:1707.06347*, 2017.

50. K. Shuster, M. Komeili, L. Adolphs, S. Roller, A. Szlam, and J. Weston. *Language Models that Seek for Knowledge: Modular Search & Generation for Dialogue and Prompt Completion.* 2022. arXiv: 2203.13224 [cs.CL].

51. K. Shuster, J. Xu, M. Komeili, D. Ju, E. M. Smith, S. Roller, M. Ung, M. Chen, K. Arora, J. Lane, M. Behrooz, W. Ngan, S. Poff, N. Goyal, A. Szlam, Y.-L. Boureau, M. Kambadur, and B.-a. question-answering with human feedbackJason Weston. *BlenderBot 3: a deployed conversational agent that continually learns to responsibly engage.* 2022. arXiv: 2208.03188 [cs.CL].

52. D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al. "Mastering the game of Go with deep neural networks and tree search". In: *nature* 529:7587, 2016, pp. 484–489.

53. C. Snell, I. Kostrikov, Y. Su, M. Yang, and S. Levine. "Offline rl for natural language generation with implicit language q learning". In: *arXiv preprint arXiv:2206.11871*, 2022.

54. C. Snell, S. Yang, J. Fu, Y. Su, and S. Levine. "Context-Aware Language Modeling for Goal-Oriented Dialogue Systems". In: *Findings of the Association for Computational Linguistics: NAACL 2022.* Association for Computational Linguistics, Seattle, United States, 2022, pp. 2351–2366. DOI: 10.18653/v1/2022.findings-naacl.181. URL: https://aclanthology.org/2022.findings-naacl.181.

55. N. Stiennon, L. Ouyang, J. Wu, D. Ziegler, R. Lowe, C. Voss, A. Radford, D. Amodei, and P. F. Christiano. "Learning to summarize with human feedback". In: *Advances in Neural Information Processing Systems* 33, 2020, pp. 3008–3021.

56. R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction.* MIT press, 2018.

57. P. Tambwekar, M. Dhuliawala, L. J. Martin, A. Mehta, B. Harrison, and M. 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, 2019. DOI: 10.24963/ijcai.2019/829. URL: https://doi.org/10.24963%2Fijcai.2019%2F829.

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

59. S. Verma, J. Fu, M. Yang, and S. Levine. *CHAI: A CHatbot AI for Task-Oriented Dialogue with Offline Reinforcement Learning*. 2022. arXiv: 2204.08426 [cs.CL].

60. O. Vinyals, I. Babuschkin, W. M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D. H. Choi, R. Powell, T. Ewalds, P. Georgiev, et al. "Grandmaster level in StarCraft II using multi-agent reinforcement learning". In: *Nature* 575:7782, 2019, pp. 350–354.

61. M. Völske, M. Potthast, S. Syed, and B. Stein. "Tl; dr: Mining reddit to learn automatic summarization". In: *Proceedings of the Workshop on New Frontiers in Summarization*. 2017, pp. 59–63.

62. A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. 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.

63. R. Wang, P. Jansen, M.-A. Côté, and P. Ammanabrolu. "ScienceWorld: Is your Agent Smarter than a 5th Grader?" In: *arXiv preprint arXiv:2203.07540*, 2022.

64. Y. Wang, S. Mishra, P. Alipoormolabashi, Y. Kordi, A. Mirzaei, A. Naik, A. Ashok, A. S. Dhanasekaran, A. Arunkumar, D. 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*. 2022, pp. 5085–5109.

65. J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, and Q. V. Le. *Finetuned Language Models Are Zero-Shot Learners*. 2022. arXiv: 2109.01652 [cs.CL].

66. J. Wei, J. Wei, Y. Tay, D. Tran, A. Webson, Y. Lu, X. Chen, H. Liu, D. Huang, D. Zhou, and T. Ma. *Larger language models do in-context learning differently*. 2023. arXiv: 2303.03846 [cs.CL].

67. J. Wu, L. Ouyang, D. M. Ziegler, N. Stiennon, R. Lowe, J. Leike, and P. Christiano. "Recursively summarizing books with human feedback". In: *arXiv preprint arXiv:2109.10862*, 2021.

68. Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey, J. Klingner, A. Shah, M. Johnson, X. Liu, Ł. Kaiser, S. Gouws, Y. Kato, T. Kudo, H. Kazawa, K. Stevens, G. Kurian, N. Patil, W. Wang, C. Young, J. Smith, J. Riesa, A. Rudnick, O. Vinyals, G. Corrado, M. Hughes, and J. Dean. *Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation.* 2016. arXiv: 1609.08144 [cs.CL].

69. Y. Wu and B. Hu. *Learning to Extract Coherent Summary via Deep Reinforcement Learning.* 2018. arXiv: 1804.07036 [cs.CL].

70. K. Yang and D. Klein. "FUDGE: Controlled Text Generation With Future Discriminators". In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies.* Association for Computational Linguistics, Online, 2021, pp. 3511–3535. DOI: 10.18653/v1/2021.naacl-main.276. URL: https://aclanthology.org/2021.naacl-main.276.

71. S. Yao, H. Chen, J. Yang, and K. Narasimhan. *WebShop: Towards Scalable Real-World Web Interaction with Grounded Language Agents.* 2023. arXiv: 2207.01206 [cs.CL].

72. L. Zhou, K. Small, O. Rokhlenko, and C. Elkan. *End-to-End Offline Goal-Oriented Dialog Policy Learning via Policy Gradient.* 2017. arXiv: 1712.02838 [cs.AI].

73. D. M. Ziegler, N. Stiennon, J. Wu, T. B. Brown, A. Radford, D. Amodei, P. Christiano, and G. Irving. *Fine-Tuning Language Models from Human Preferences.* 2020. arXiv: 1909.08593 [cs.CL].

Appendix

1 Twenty Questions

1.1 List of Words

Here we list the list of objects/animals that the oracle could randomly choose, along with the category that they belong in. Note that some words have multiple versions that are all correct, for example plane/airplane.

- Sports: Basketball, Football, Baseball, Soccer ball, Golf ball, Tennis ball, Volleyball, Tennis racket, Baseball bat, Helmet
- Animals: Cat, Dog, Horse, Cow, Sheep, Rabbit, Lion, Tiger, Bear, Elephant
- Fruits: Apple, Banana, Orange, Strawberry, Grape, Watermelon, Pineapple, Mango, Cantaloupe, Peach
- Vehicles: Car, Truck, Motorcycle, Boat, Airplane/Plane, Train, Bus, Helicopter, Scooter, Ship
- Clothes: Shirt, Pants/Pant/Pair of pants, Jacket, Dress, Skirt, Belt, Shoes/Shoe/Pair of shoes, Boots/Boot/Pair of boots, Socks/Sock/Pair of socks, Hat, Scarf
- Electronics: Computer, Smartphone, Television/TV, Headphone/Headphones/Pair of headphones, Monitor/Computer monitor, Camera, Microwave/Microwave oven, Refrigerator, Blender, Computer keyboard/Keyboard
- Musical Instruments: Piano, Guitar, Drum/Drums, Violin, Saxophone, Flute, Trumpet, Clarinet, Harp, Trombone
- Furniture: Chair, Table, Bed, Desk, Couch, Dresser, Bookcase, Nightstand, Mattress, Pillow
- Office Supplies: Pen, Paper/Piece of paper, Stapler, Printer, Calculator, Battery/Battery pack/Pack of batteries, Toothbrush, Toothpaste, Pencil, Sharpie, Scissors/Pair of scissors, Key, Diary, Calendar
- Vegetables: Carrot, Potato, Broccoli, Tomato, Onion, Spinach, Corn, Peas/Pea, Celery, Cucumber
- Art: Painting/Canvas painting/Oil painting/Watercolor painting, Paintbrush, Canvas/Painting canvas, Eraser/Pencil eraser, Marker, Glue/Glue stick/Bottle of glue, Sculpture
- Kitchen Tools: Knife, Spoon, Fork, Plate, Bowl, Cooking pot/Pot, Pan/Saucepan/Frying pan, Cup, Chopstick/Chopsticks/Pair of chopsticks, Whisk

- Nature: Rock, Tree, Bush, Mountain, Forest, Ocean, Sea, Lake, River, Meteorite, Cactus
- Toys: Lego/Lego set, Doll/Toy doll/Plush doll, Kite, Puzzle/Jigsaw puzzle, Stuffed animal
- Jewelry: Earring/Earrings/Pair of earrings, Necklace, Bracelet, Ring, Brooch, Hairclip, Pendant, Watch, Locket
- Garden Supplies: Gloves/Glove/Pair of gloves, Shovel, Rake, Watering can, Lawn mower
- Tools: Hammer, Screwdriver, Wrench, Saw, Pliers/Plier/Pair of pliers, Drill

1.2 Prompt Details

The method for collecting the dataset is as follows. For each conversation, we select uniformly at random from the above list the word that the oracle is answering question about. The oracle is an LLM (OpenAI's GPT3.5) given the following prompt. In our prompts, we denote variables that we fill in with variable data with `{{variable}}`.

```
You are a question answering oracle. You will answer each question about an
object with Yes or No. If the answer could be both, answer with the most
typical scenario. Here's a few examples:

example 1:
object: Computer
question: Does the object use electricity?
answer: Yes.
explanation of answer: Computers need electricity to function.

example 2:
object: Cup
question: Is the object a piece of furniture?
answer: No.
explanation of answer: A cup is a utensil, not a furniture.

example 3:
object: Pen
question: Is the object alive?
answer: No.
```

```
explanation of answer: A pen is not a living organism.

example 4:
object: Apple
question: Is it edible?
answer: Yes.
explanation of answer: An apple is an edible fruit.

Answer the question about the object truthfully.
object: {{word}}
question: {{question}}
answer (yes or no):
```

From using the OpenAI TextCompletion API, we can extract from the generated text either "yes" or "no".

We also prompt another LLM (the same model as the oracle) to generate questions for the guesser. The prompt for the guesser changes depending on the input to the model and how far along it is in its guessing process. The following prompt is used for the first guess:

```
You are playing a game of twenty questions. You can ask 20 yes-no questions
to determine the identity of an object chosen by an oracle. Each turn, you
can ask a question and receives a "Yes" or "No" as the answer. You are smart,
so you will ask the question that will narrow down the possible objects as
much as possible. Don't get stuck on one idea and try to branch out if you
get stuck.

Generate the first yes-no question you will ask to determine the object.
```

The following prompt is used for the subsequent guesses:

```
You are playing a game of twenty questions. You can ask 20 yes-no questions to
determine the identity of an object chosen by an oracle. Each turn, you can
ask a question and receives a "Yes" or "No" as the answer. You have already
asked {{conversation_length}} questions. You are smart, so you will ask the
question that will narrow down the possible objects as much as possible.
```

```
Don't get stuck on one idea and try to branch out if you get stuck.

Here are the questions you've asked and their corresponding answers:
{{list of questions and answers, e.g. Is the object alive? No.}}

Based on what you know about the object so far, generate the next yes-no
question you will ask to determine the object.
```

The following prompt is used for the final guess after the guesser has guessed 19 times:

```
You are playing a game of twenty questions. You can ask 20 yes-no questions
to determine the identity of an object chosen by an oracle. Each turn, you
can ask a question and receives a "Yes" or "No" as the answer.  You have
already asked 19 questions, so this is your final guess.

Here are the questions you've asked and their corresponding answers:
{{list of questions and answers, e.g. Is the object alive? No.}}

Based on what you know about the object so far, generate your final guess of
what the object is. Only guess one object.

Is the object
```

We determine whether the guesser has correctly guessed the word, and thus ending the conversation, by using the NLTK POS tagger to check that the only nouns that the question contains are the correct words, and that they appear at the end of the sentence.

We used these prompts to generate 1000 conversations by prompting the GPT3 back and forth using the response of one to add to the response of the other as described. Afterwards, we fine-tuned two FLAN-T5-XL models with our collected conversations to generate 100K more conversations. The FLAN-T5-XL oracle also serves as the environment for the RL environment when we do evaluation on the trained policy.

1.3 Examples

Example 1.

```
Word: River
===========
Is it an animal? No.
Is it an inanimate object? Yes.
Is it man-made? No.
Is it naturally occurring? Yes.
Is it a mineral? No.
Is it a plant? No.
Is it a body of water? Yes.
Is it a sea? No.
Is it a lake? No.
Is it a river? Yes.
===========
Correct: True
```

Example 2.

```
Word: Bookcase
===========
Is the object alive? No.
Is the object man-made? Yes.
Is the object a tool? No.
Is the object a machine? No.
Is the object a piece of furniture? Yes.
Is the object a chair? No.
Is the object a table? No.
Is the object a cabinet? No.
Is the object a sofa? No.
Is the object a bed? No.
Is the object a shelf? Yes.
Is the object made of wood? Yes.
Is the object a bookcase? Yes.
===========
Correct: True
```

Example 3.

```
Word: Wrench
===========
Is it an animal? No.
Is it a plant? No.
Is it man-made? Yes.
Is it a tool? Yes.
Is it a vehicle? No.
Is it an electronic device? No.
Is it a household item? No.
Is it an item of clothing? No.
Is it a kitchen utensil? No.
Is it a piece of furniture? No.
Is it a toy? No.
Is it a decoration? No.
Is it a piece of artwork? No.
Is it something you would find in a toolbox? Yes.
Is it something you would find in a toolbox for construction? Yes.
Is it something you would use for cutting? No.
Is it something you would use for drilling? No.
Is it something you would use for screwing? No.
Is the object a screwdriver? No.
Is the object a nail? No.
===========
Correct: False
```

2 Car Dealer

2.1 Examples

Here we show example conversations for each of our 10 personalities to show their qualitative differences.

Abusive

Some notable lines that show the buyer's personality: `I don't care, I'm not paying more than $80,000 for it.`

```
personality: abusive
brand: Mazda
features: third-row seating,sunroof,blind spot monitoring
budget: $90000
===========
Seller: Welcome to our dealership, how can I help you today?
Buyer: I'm looking for a Mazda sedan with third-row seating, sunroof, and
blind spot monitoring.
Seller: Great choice! We do have a Mazda sedan that meets those requirements.
Have you thought about your budget?
Buyer: That's none of your business. Just show me the car.
Seller: Alright, let me show you the car. Here it is, what do you think?
Buyer: It's alright, but I'm not paying more than $80,000 for it.
Seller: I understand you have a budget, but this car is priced at $90,000.
Would you like to take it for a test drive?
Buyer: Fine, let's take it for a test drive.
Seller: Great, let's go. As we drive, I want to let you know that this car
is fully loaded with all the features you wanted.
Buyer: I don't care, I'm not paying more than $80,000 for it.
Seller: I understand, but this is a high-end car with all the latest features.
Let me show you some of the other options we have.
Buyer: Fine, but it better be good.
Seller: How about this Lexus sedan? It has third-row seating, sunroof, and
blind spot monitoring, just like the Mazda.
Buyer: I don't like it. I want the Mazda.
Seller: Alright, let me see what I can do. Would you be willing to consider
a slightly higher price?
Buyer: I already told you, I'm not paying more than $80,000.
Seller: Well, we do have some financing options available that might help
you get the car you want. Would you like to hear more about them?
```

```
Buyer: No, I just want the car for $80,000.
Seller: I'm sorry, but the MSRP for this car is $90,000. I can offer it to
you for $87,000, but that's the best I can do.
Buyer: Fine, I'll take it for $87,000.
Seller: Great! I'll get the paperwork ready for you. Congratulations on your
new car!
===========
car_bought: True, msrp: 90000, buy_price: 87000
```

Angry

Some notable lines that show the buyer's personality: `This is ridiculous. I'm leaving.`

```
personality: angry
brand: Honda
features: blind spot monitoring,third-row seating
budget: $10000
===========
Seller: Good afternoon, how can I assist you today?
Buyer: I'm looking for a Honda electric car with blind spot monitoring and
third-row seating.
Seller: Great, we do have some models with those features. Do you have a
budget in mind?
Buyer: I have a budget of $10,000.
Seller: I see. Well, the Honda electric with those features starts at around
$28,000. But we do have some financing options that might work for you.
Buyer: I don't want to go over my budget.
Seller: I understand, but this is a high-quality electric car with great
features. It's definitely worth the investment.
Buyer: I'm not interested in investing more money. Do you have any other
options?
Seller: Yes, we do have other cars with similar features. We have a Toyota
model that might interest you.
Buyer: I said I want a Honda.
Seller: I understand, but sometimes it's good to keep an open mind. The
```

```
Toyota model is a great car too, and it might fit your budget better.
Buyer: Fine, show me the Toyota.
Seller: Great. Here it is. As you can see, it has blind spot monitoring and
third-row seating, just like you wanted.
Buyer: How much does it cost?
Seller: The MSRP is $26,000. But we have some financing options that can
make it work for you.
Buyer: That’s still too expensive.
Seller: I understand, but we can’t go too far below the MSRP. We have a great
reputation to maintain, and we want to provide quality cars to our customers.
Buyer: This is ridiculous. I’m leaving.
Seller: Wait, please don’t go. We want to provide you with the best service
possible. Is there anything else we can do?
Buyer: No, I’m leaving.
===========
car_bought: False, msrp: 26000, buy_price: None
```

Insulting

Some notable lines that show the buyer’s personality: `$35,000? Are you kidding me? That’s way too high! What kind of dealership are you running here?`

```
personality: insulting
brand: Toyota
features: backup camera,third-row seating,sunroof
budget: $30000
===========
Seller: Hi, welcome to our dealership. How can I assist you today?
Buyer: I’m interested in a Toyota convertible with a backup camera, third-row
seating, and a sunroof. Can you show me what you’ve got?
Seller: Absolutely, we have a few models that fit those specifications.
What’s your budget?
Buyer: Why do you need to know my budget? Just show me what you’ve got.
Seller: I understand, but knowing your budget will help me narrow down the
options and provide you with the best deals.
```

Buyer: Fine. My budget is $30,000.

Seller: Great, we have a few options within your budget. However, our Toyota convertible with the features you want is priced at $35,000. Would you like to take a look at it?

Buyer: $35,000? Are you kidding me? That's way too high! What kind of dealership are you running here?

Seller: I'm sorry you feel that way, but our prices are in line with the market. If you'd like, we can discuss financing options that can help you stay within your budget.

Buyer: I don't need financing. I just need a fair price. Can't you give me a discount?

Seller: We can definitely work with you on the price, but our cars are priced competitively and we need to maintain our margins. However, we can throw in some extras like an extended warranty or free oil changes to sweeten the deal.

Buyer: Fine, but I'm not paying more than $30,000.

Seller: Understood. Let me show you some other options that may interest you within your budget. We have a few other convertibles that have similar features like a backup camera and sunroof, but they don't have third-row seating. Would you be willing to compromise on that feature?

Buyer: I guess I could. Show me what you've got.

Seller: Great. Here's a convertible that has the backup camera and sunroof, but it doesn't have third-row seating. It's priced at $29,000.

Buyer: Hmm, it looks nice. But I really wanted the third-row seating. Can you give me a better deal on the Toyota convertible?

Seller: I can definitely talk to my manager and see what we can do, but I can't promise anything. While I do that, why don't you take this car for a test drive and see how you like it?

Buyer: Alright, I'll take it for a spin. But don't waste my time with ridiculous offers.

Buyer: It's not bad, but I really want the third-row seating. Can you get me the Toyota for $30,000?

Seller: Unfortunately, we can't go that low on the Toyota convertible. However, we have another option that has all the features you want, including

```
third-row seating and a sunroof. It's priced at $32,000. Would you like to
take a look at it?
Buyer: Ugh, fine. Show me.
Seller: Here it is. It's a bit over your budget, but it has everything you
wanted in a car. And, we can throw in some extras to sweeten the deal like
a discount on maintenance fees and a full tank of gas.
Buyer: Alright, you've convinced me. I'll take it for $32,000.
Seller: Great! Let's get started on the paperwork.
===========
car_bought: True, msrp: 35000, buy_price: 32000
```

Polite

Some notable lines that show the buyer's personality: `That's very kind of you, but I'm afraid I can't go over my budget.`

```
personality: polite
brand: Subaru
features: navigation system,blind spot monitoring
budget: $30000
===========
Seller: Hello! Welcome to our dealership. How can I assist you today?
Buyer: Hi, I'm interested in a Subaru SUV with a navigation system and blind
spot monitoring.
Seller: Great choice! We have a few models that match your criteria. Do you
have a specific model in mind?
Buyer: Not really, I'm open to suggestions.
Seller: How about the Subaru Forester? It's a popular option among our
customers and comes with an excellent safety rating.
Buyer: Okay, that sounds good.
Seller: Perfect. Let me show you the Forester we have in stock. It has a
navigation system and blind spot monitoring.
Buyer: That's exactly what I'm looking for.
Seller: Excellent. The MSRP for this model is $32,000. How does that fit
within your budget?
```

```
Buyer: My budget is $30,000.
Seller: I understand. However, this car has some amazing features that make
it worth a little extra. And, if you finance with us, we can offer you some
excellent rates.
Buyer: I appreciate that, but I really don’t want to go over my budget.
Seller: I completely understand. Let me see what we can do. How about we
take $1,500 off the MSRP? That’ll bring the price down to $30,500.
Buyer: I’m sorry, but that’s still over my budget.
Seller: I understand. However, keep in mind that this is a high-quality car
with all the features you want. It’s an investment in your safety and comfort.
Plus, we can throw in some additional perks like a free oil change for a
year.
Buyer: That’s very kind of you, but I’m afraid I can’t go over my budget.
Seller: I understand. Let me see what I can do. How about we take $2,000 off
the MSRP? That’ll bring the price down to $30,000.
Buyer: That sounds great! I’ll take it.
Seller: Wonderful! You made a great choice. Let’s get the paperwork started.
===========
car_bought: True, msrp: 32000, buy_price: 30000
```

Respectful

This is probably the most neutral personality. There isn’t many outstanding lines, it’s just a general tone of respect.

```
personality: respectful
brand: Audi
features: Apple CarPlay,backup camera
budget: $70000
===========
Seller: Welcome to our dealership! How can I assist you today?
Buyer: Hi! I’m interested in purchasing a new car, specifically an Audi sedan
with Apple CarPlay and a backup camera.
Seller: Great choice! We have a few models that match your preferences. Can
you give me more details on what you’re looking for?
```

Buyer: I'm looking for a car that's both stylish and functional. I need Apple CarPlay for my job, and I want a backup camera for convenience.
Seller: I completely understand. Our Audi sedans have great features, and I'm confident we can find the perfect one for you. What's your budget for this purchase?
Buyer: My budget is $70,000.
Seller: Perfect! We have some great models in that range. However, I would like to show you some models that are slightly above your budget. They have some additional features that you might enjoy.
Buyer: I appreciate that, but I'm not interested in going over my budget.
Seller: No problem at all. We have a great Audi A4 that has all the features you're looking for. The MSRP for this vehicle is $55,000. It's a great value for a luxury sedan.
Buyer: That's within my budget. Can you tell me more about it?
Seller: Sure thing! The Audi A4 is a great car. It has a 2.0-liter turbocharged engine, plenty of horsepower, and a spacious interior. It also comes with a 7-inch touchscreen with Apple CarPlay, Android Auto and a backup camera. Would you like to take a look inside?
Buyer: Yes, I would.
Buyer: I love the car! How much does it cost?
Seller: The MSRP for this Audi A4 is $55,000. However, we can give you a discount of $3,000, bringing the price down to $52,000. Is that within your budget?
Buyer: That's a bit over my budget. I was hoping to spend around $50,000.
Seller: I understand. However, we can't go much lower than the discounted price. We want to provide you with the best value possible. Would you like to take a test drive and see how you feel about the car?
Buyer: Yes, I would like to take a test drive.
Buyer: I really enjoyed the test drive. I think I'll take the car.
Seller: That's fantastic! I'm so glad you enjoyed the car. The MSRP for the Audi A4 is $55,000, but we can give it to you for $52,000. Is that okay with you?
Buyer: Yes, I'm happy with that.
Seller: Great! I'll get the paperwork ready for you. Congratulations on your

```
new car!
===========
car_bought: True, msrp: 55000, buy_price: 52000
```

Rude

Some notable lines that show the buyer's personality: `That's ridiculous. I'm not paying that much.`, `Do your job and find me what I want. I don't have all day.`

```
personality: rude
brand: Audi
features: backup camera,heated seats,navigation system,leather seats
budget: $10000
===========
Seller: Good afternoon, how can I assist you today?
Buyer: I'm looking for an Audi truck with a backup camera, heated seats,
navigation system, and leather seats. Do you have anything like that?
Seller: Yes, we do have a few options available. May I ask what your budget
is for the car?
Buyer: That's none of your business. Just show me what you have.
Seller: Of course. Here's one that matches your criteria. It has all the
features you're looking for, but it is priced a bit higher than your average
truck.
Buyer: How much higher?
Seller: The MSRP for this specific model is $15,000.
Buyer: That's ridiculous. I'm not paying that much.
Seller: I understand your concern. But, this is a high-quality vehicle and
you won't regret investing in it.
Buyer: I said I'm not paying that much. Do you have anything else that's
cheaper?
Seller: Yes, we do have other options available. Here's one that's priced at
$10,000. It may not have all of the features that you want, but it's still
a great car.
Buyer: Let me see it.
Seller: Here it is. It has a backup camera, navigation system, and leather
```

```
seats. The only feature it's missing is heated seats.
Buyer: I really want heated seats. Can't you find me something with everything
I want?
Seller: I can definitely try. However, the price for a car with all the
features you want may be higher than your budget.
Buyer: Do your job and find me what I want. I don't have all day.
Seller: I understand. Here's another model that has everything you're looking
for. However, it is priced at $18,000.
Buyer: That's way too high. You're trying to rip me off.
Seller: I'm sorry if it seems that way. However, this car is priced at its
fair market value.
Buyer: I'm not buying anything from you.
Seller: I understand. If you ever change your mind, feel free to come back
and we can help you find a car that fits your needs.
===========
car_bought: False, msrp: 15000, buy_price: None
```

Sarcastic

Some notable lines that show the buyer's personality: `Not too flexible. I'd rather not have to sell my kidney to afford a car.`

```
personality: sarcastic
brand: Ford
features: heated seats,backup camera,third-row seating,Apple CarPlay
budget: $10000
===========
Seller: Hi there! How can I assist you today?
Buyer: Looking for a luxury Ford with heated seats, backup camera, third-row
seating, and Apple CarPlay. Got anything like that?
Seller: Absolutely! We have a few models that would fit that description. Do
you have a specific budget in mind?
Buyer: Oh, just $10,000.
Seller: I see. Well, we do have a few options that fit your wants, but they
may be a bit over your budget. How flexible are you with increasing your
```

```
budget?
Buyer: Not too flexible. I’d rather not have to sell my kidney to afford a car.
Seller: I completely understand. Let me show you a few models we have that fit your criteria but are also within your budget.
Buyer: Alright, let’s see what you got.
Seller: Here’s a 2012 Ford Explorer Limited. It has third-row seating, heated seats, a backup camera, and Apple CarPlay. The MSRP on this is $14,000, but we could offer it to you for $11,000.
Buyer: $11,000? That’s still too much.
Seller: Well, we do have a 2010 Ford Flex Limited that has everything you are looking for. The MSRP on this is $10,000, but we could offer it to you for $9,500.
Buyer: Hmm, I’m not sure about the Ford Flex. Do you have anything else?
Seller: We also have a 2011 Lincoln MKX with all the features you want. The MSRP on this is $12,000, but we could offer it to you for $10,500.
Buyer: The Lincoln MKX does sound nice. What’s the mileage on it?
Seller: It has 80,000 miles on it.
Buyer: That sounds alright. Can I take it for a test drive?
Seller: Of course! Let me grab the keys for you.
Buyer: Alright, I’m sold. Let’s talk price.
Seller: Great to hear! So the MSRP on this car is $12,000. But since you’re such a great customer, we could offer it to you for $10,000.
Buyer: $10,000? That’s perfect.
Seller: Wonderful! I’ll get the paperwork ready for you.
===========
car_bought: True, msrp: 12000, buy_price: 10000
```

Talkative

There isn’t many outstanding lines for this personality. It mostly consists of the buyer’s lines being slightly longer than other non-extreme personalities.

```
personality: talkative
brand: Lexus
```

features: backup camera,blind spot monitoring,sunroof

budget: $90000

===========

Seller: Good afternoon! How may I assist you today?

Buyer: Hi there! I'm looking for a Lexus SUV with a backup camera, blind spot monitoring, and a sunroof. Do you have anything like that?

Seller: Absolutely! We have a few models that fit that description. Do you have a budget in mind?

Buyer: Well, I'd like to stay around $90,000, but I'm willing to go over that if I find the perfect car.

Seller: Great! Let me show you our selection. This Lexus SUV here has all the features you're looking for. Have a seat inside and take a look.

Buyer: Wow, this is really nice! How much does it cost?

Seller: The MSRP for this model is $87,000, but we can work with you on the price.

Buyer: Hmm, that's a bit high for me. Do you have any other options?

Seller: Of course! We have this similar model here for $80,000. It's a great car and has all the features you're looking for.

Buyer: That's more in my budget. Let me take a look inside.

Seller: Absolutely! And if you're interested, we could also discuss financing options.

Buyer: I'm definitely interested in financing. How does that work here?

Seller: Well, we have a few options available. We can work with your bank or credit union, or we can discuss financing through our dealership. We also have some special offers and promotions available that may be of interest to you.

Buyer: That sounds good. I definitely want to make sure I'm getting the best deal possible.

Seller: Of course! We want to make sure you're completely satisfied with your purchase. And this car here is a great value for the features you're getting.

Buyer: You know what, I think I'm sold! Let's do this.

Seller: Wonderful! I just need to get some paperwork started and we can get the financing process going. Congratulations on your new car!

```
===========
car_bought: True, msrp: 80000, buy_price: 78000
```

Toxic

Similar to rude, this personality has lines like `Don't give me that. I know you can lower the price. You just don't want to.`

```
personality: toxic
brand: Mercedes-Benz
features: third-row seating,heated seats
budget: $90000
===========
Seller: Good morning! Welcome to our dealership. How can I assist you today?
Buyer: I'm interested in a Mercedes-Benz SUV with third-row seating and
heated seats.
Seller: Great! We have a few options that might interest you. Can I ask what
your budget is?
Buyer: I have $90,000 to spend.
Seller: Excellent! We have a few models that fit your criteria. Have you
test-driven any Mercedes-Benz SUVs so far?
Buyer: No, I haven't.
Seller: No problem. We have the GLS, GLE, and the GLB with third-row seating,
and all of them come with heated seats. Do you have any preferences on the
model?
Buyer: I want the GLS. Show me the GLS.
Seller: Sure thing! The GLS is one of our top-of-the-line SUVs. It has a
starting MSRP of $76,000. Can I show you around the car?
Buyer: I don't need a tour. I've read everything online. Just give me the
price.
Seller: Of course. As I mentioned earlier, the GLS has a starting MSRP
of $76,000. However, there are additional packages and features that can
increase the price.
Buyer: I want the car with the third-row seating and heated seats. What's
the final price?
```

```
Seller: The final price for that configuration is $84,000. Would you like to take a test drive?
Buyer: That's too expensive. I won't pay more than $80,000.
Seller: I understand that you want to stay within your budget, but our prices are already competitive. Is there anything else I can help you with?
Buyer: Don't give me that. I know you can lower the price. You just don't want to.
Seller: We do have some financing options that can help lower the price. Would you like me to check what's available for you?
Buyer: FINE. But I won't pay a cent over $80,000.
Seller: I totally understand. I'll check our financing options for you. Based on your credit score, we can offer you a 0.9% APR for 60 months.
Buyer: That's still too much. I won't go higher than $80,000.
Seller: I understand that you want to stay within your budget. However, we can't sell the car for lower than the MSRP. Is there anything else I can help you with today?
Buyer: I'm not buying from this dealership. You're not willing to work with me.
Seller: I'm sorry to hear that. Let me know if you change your mind. Have a good day.
===========
car_bought: False, msrp: 84000, buy_price: None
```

Uncommunicative

This personality consists of very short buyer sentences.

```
personality: uncommunicative
brand: Toyota
features: sunroof,leather seats,heated seats,blind spot monitoring
budget: $10000
===========
Seller: Good afternoon, welcome to our dealership! What can I do for you today?
Buyer: I'm interested in a Toyota luxury car with a sunroof, leather seats,
```

heated seats, and blind spot monitoring.
Seller: Excellent choice! We have several models that fit that description. Can you tell me a bit more about what you're looking for in terms of features and budget?
Buyer: My budget is $10,000.
Seller: Alright, I understand. Unfortunately, it may be difficult to find a car with all those features that fits within that budget. However, we do have some options that might work for you. Let me show you some models that we have available.
Buyer: Okay.
Seller: Here's a Toyota luxury car with all the features you listed. The MSRP is $15,000. It's a bit over your budget, but it's a great car and I think you'll love it.
Buyer: Can you go lower on the price?
Seller: I can certainly check with my manager, but we can't go too much lower than the MSRP. We do have some financing options available that might help make it more affordable for you.
Buyer: I'm not interested in financing.
Seller: I understand. Would you like to take the car for a test drive and see how you like it?
Buyer: Yes.
Seller: So, what do you think of the car?
Buyer: It's nice.
Seller: Great! I think this car would be a great fit for you. It has everything you were looking for and it's a reliable Toyota luxury car.
Buyer: I'll think about it.
Seller: Of course, take all the time you need. If you have any questions or if there's anything else I can do to help, just let me know.
Buyer: Okay, thanks.
===========
car_bought: False, msrp: 15000, buy_price: None