Title
Towards Understanding How Machines Can Learn Causal Overhypotheses

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 45(45)

Authors
Kosoy, Eliza
Chan, David M
Liu, Adrian
et al.

Publication Date
2023

Peer reviewed

Towards Understanding How Machines Can Learn Causal Overhypotheses

Eliza Kosoy*[,1], David M. Chan*[,1], Adrian Liu[1], Jasmine Collins[1], Bryanna Kaufmann[1], Sandy Han Huang[2], Jessica B. Hamrick[2], John Canny[1], Nan Rosemary Ke[2], Alison Gopnik[1]

Abstract

Recent work in machine learning and cognitive science has suggested that understanding causal information is essential to the development of intelligence. One of the key challenges for current machine learning algorithms is modeling and understanding *causal overhypotheses*: transferable abstract hypotheses about sets of causal relationships. In contrast, even young children spontaneously learn causal overhypotheses, and use these to guide their exploration or to generalize to new situations. This has been demonstrated in a variety of cognitive science experiments using the "blicket detector" environment. We present a causal learning benchmark adapting the "blicket" environment for machine learning agents and evaluate a range of state-of-the-art methods in this environment. We find that although most agents have no problem learning causal structures seen during training, they are unable to learn causal *overhypotheses* from these experiences, and thus cannot generalize to new settings.

Keywords: causal learning; causal overhypotheses; blicket detector; causal benchmarks

Introduction

Research in causal modeling has long studied not only the accurate modeling of in-distribution data, but also the accurate recovery of underlying causal mechanisms (and their true graphical relations) capable of explaining *out-of-distribution* data, opening the way for models achieving systematic generalization (Bengio et al., 2019; Schölkopf et al., 2021; Ke et al., 2021).

One of the key components of causal learning are causal overhypotheses: a method for describing priors over stochastic causal graphs (Kemp, Perfors, & Tenenbaum, 2007; Kemp, Goodman, & Tenenbaum, 2010; Lucas & Griffiths, 2010; Perfors, Tenenbaum, Griffiths, & Xu, 2011; Tenenbaum, Kemp, Griffiths, & Goodman, 2011; Gopnik & Wellman, 2012). Causal overhypotheses enable models (and humans) to learn from sparse data (Griffiths & Tenenbaum, 2009) by reducing the likely set of possible causal relationships in a graph. Despite the recent surge of machine learning datasets and environment for causal inference and learning (Ahmed et al., 2020; McDuff et al., 2021; J. X. Wang et al., 2021; Ke et al., 2021), causal overhypotheses for these environments and datasets are unclear, rendering existing benchmarks unable to evaluate the competency of existing agents when learning and using causal overhypotheses.

In this work we seek to fill this gap by leveraging a recent benchmark environment (Kosoy et al., 2022) drawing inspiration from recent cognitive science work using blicket detectors (Gopnik & Sobel, 2000; Lucas, Bridgers, Griffiths, & Gopnik, 2014), and evaluating this benchmark for the first time on several state of the art causal models. We find that such algorithms, in contrast to children, as explored by Kosoy et al. (2022), only converge on a solution after an extensive number of trials or if they are given all the possible settings and outcomes as training data. This suggests that these tasks are an interesting challenge for machine learning algorithms. In order for machines to perform as well as children do, algorithms must reason about the sequence of observations seen, extract causal overhypotheses from those observations and use them for exploration—which current methods fall short of doing.

Related Work

Blicket Detectors for Causal Learning A "blicket detector" is a machine that lights up and plays music when some combinations of objects but not others are placed on it (Gopnik & Sobel, 2000; Lucas et al., 2014). The central question is whether an agent can learn that a particular set of causal events will lead to the lighting-up effect, and use that knowledge to design novel interventions on the machine. The causal relationship is entirely determined by the pattern of conditional dependencies and interventions, rather than requiring intuitive physics knowledge or visual understanding.

Several features of this environment and the tasks it allows make it particularly useful as a benchmark for machine learning algorithms. First, causal representations are more powerful and structured than mere statistical generalizations, though both can be systematically inferred from statistical information. Many researchers (e.g. Pearl, Spirtes et al., Bengio) have argued that such causal representations are crucial for both human and general AI intelligence. Second, unlike some existing causal environments (Ke et al., 2021; J. X. Wang et al., 2021) the blicket environment enables the inference of both *specific causal structure* and more *general features of causal structure*, such as whether causal systems are conjunctive or disjunctive, stochastic or deterministic. Learning these *overhypotheses* about causal structure (Griffiths & Tenenbaum, 2009) is especially important because such inferences can constrain the search for causal structure, a search that can rapidly become intractible.

Most significantly, the environment allows for a direct comparison to human agents. The work of Kosoy et al. (2022) first described a version of a blicket environment, and demonstrated that even preschool children can easily manipulate

* Authors contributed equally, [1] University of California, Berkeley [2] Deepmind, Corresponding authors: eko@berkeley.edu, davidchan@berkeley.edu

In M. Goldwater, F. K. Anggoro, B. K. Hayes, & D. C. Ong (Eds.), *Proceedings of the 45th Annual Conference of the Cognitive Science Society*.

and explore blicket environments, generate appropriate data, and rapidly learn both particular causal structure and overhypotheses about causal structure (Kosoy et al., 2022). In their experiments with children from ages four to six, Kosoy et al. (2022) found that children exhibit a diverse range of exploration strategies, which suggests that they are exploring based on a rich set of causal overhypotheses, formed from their prior knowledge of how objects and mechanisms behave, while simple RL models found such interactions challenging. Indeed, exploring this environment in the context of children, particularly young children leads to a singularly informative baseline group; They do not have the extensive education and experience of typical adults, which might make comparisons to artificial agents challenging, but they are nevertheless effective causal learners and able to make broad yet accurate generalizations from small sample sizes, in contrast to many current machine learning systems (Gopnik, 2012; Gopnik et al., 2017).

Causal Overhypotheses Causal overhypotheses are hierarchical priors over the structure and/or conditionals of a causal model, and have been widely studied in the cognitive science literature (Kemp et al., 2007, 2010; Lucas & Griffiths, 2010; Perfors et al., 2011; Tenenbaum et al., 2011; Gopnik & Wellman, 2012). An overhypothesis might state that the causal graph itself has a particular form (such as a "common effect" or a chain structure), or that the conditionals within that graph have a particular form (such as that $p(X|Y,Z)$ follows a particular parametric distribution). Having a good causal overhypothesis is a form of inductive bias that can make causal inferences much easier: for example, while we might not know the *specific* causal graph, if we know that it takes the form of a common effect, then we need only a few interventions (possibly only $O(N)$) to fully determine the specific causal graph. Ideally, then, machine learning agents should be able to learn such overhypotheses and leverage them to make more efficient and accurate causal inferences.

Multi-task and Causal RL Benchmarks There exist multi-task RL benchmarks featuring robotics (Yu et al., 2019; James, Ma, Arrojo, & Davison, 2020), physical reasoning (Bakhtin, van der Maaten, Johnson, Gustafson, & Girshick, 2019; Allen, Smith, & Tenenbaum, 2020), and video games (Cobbe, Klimov, Hesse, Kim, & Schulman, 2018; Machado et al., 2018; Nichol, Pfau, Hesse, Klimov, & Schulman, 2018; Chevalier-Boisvert et al., 2018). Unfortunately, since it is not clear what the relevant causal overhypotheses for these environments are, it is difficult to evaluate how causal information influences agents' exploration.

RL benchmarks for causal discovery include Causal World (Ahmed et al., 2020), Causal City (McDuff et al., 2021), Alchemy (J. X. Wang et al., 2021), ACRE (Zhang, Jia, Edmonds, Zhu, & Zhu, 2021), and the work of Ke et al. (2021). However, many of these environments either lack clear causal hypotheses or do not allow for controlling overhypotheses. In addition, these environments primarily focus on causal induction or generalization, rather than exploration (though see Sontakke, Mehrjou, Itti, and Schölkopf (2021)). In contrast, the blicket environment in this work is designed to measure agents' ability to explore using causal overhypotheses. Moreover, children have not been tested on any of these existing environments, whereas in the blicket environment, prior work has shown that children as young as age four are able to learn causal overhypotheses and use these to explore effectively (Kosoy et al., 2022). It can be informative to compare the exploration and performance of RL approaches to that of children.

Language Models for Reasoning Tasks Large language models such as GPT (Radford et al., 2019; Brown et al., 2020) and PALM (Chowdhery et al., 2022) are trained on massive amounts of data, and they have been shown to be able to express uncertainty and perform common sense reasoning up to an extent (Lin, Hilton, & Evans, 2022). In this work, we probe the the causal reasoning capabilities of GPT-3 and PALM using textual descriptions of the virtual blicket environment.

Evaluating Causal Learning in the Blicket Environment

Results from (Kosoy et al., 2020) suggest that children can explore efficiently, especially given the causal overhypotheses. In this work, we evaluate how a spectrum of different machine learning models perform on the blicket detector tasks. Solving these tasks requires reasoning about the sequence of observations seen, extracting causal overhypotheses from those observations, and using these extracted overhypotheses for exploration. In this work, evaluate several popular reinforcement learning algorithms—A2C (Mnih et al., 2016), PPO2 (Schulman, Wolski, Dhariwal, Radford, & Klimov, 2017), and Q-learning (Watkins & Dayan, 1992)—on this task. We further evaluate imitation learning algorithms, including behaviour cloning with decision transformers (Chen et al., 2021). Finally, we apply pre-trained language models (Brown et al., 2020; Chowdhery et al., 2022), since they have been shown to be capable of performing common-sense reasoning and expressing uncertainty to an extent (Lin et al., 2022).

Experimental Design

In our blicket environment, we build on the environment first explored in Kosoy et al. (2022), and consider two causal overhypotheses, shown in Figure 1: that the world is either *conjunctive* or *disjunctive*, with the causal system defined as follows . In both cases, the causal graph takes the form of a common effect, with $N+1$ variables: N objects $\mathbb{X} = \{X_0, X_1, ..., X_N\}$ (the causes) and one blicket machine M (the effect). Each object X_i can either be on top of the blicket machine ($X_i = 1$) or off the machine ($X_i = 0$). The blicket machine can be either on ($M = 1$) or off ($M = 0$). Because the graph is a common effect, objects' states do not influence each other (intervening on $X_i = 1$ does not impact X_j). Objects' states may, however,

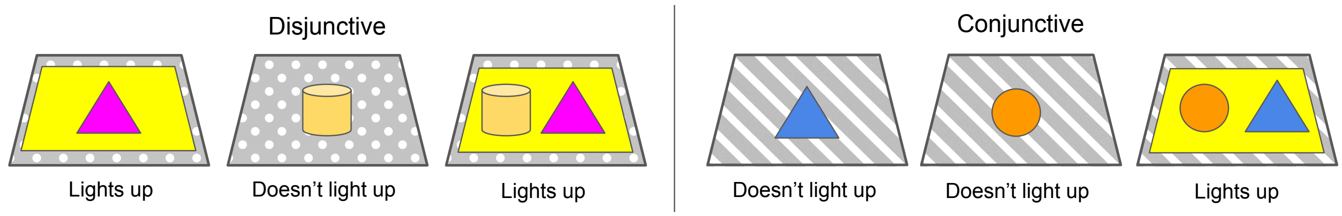


Figure 1: A simplified rendering of the virtual blicket detector environment. In the disjunctive setting (left), only one blicket is needed for the blicket to light up. Whereas in the conjuctive setting (left), two blickets are needed. These examples are shown to children for the *given hypotheses* condition in Kosoy et al. (2022).

influence the state of the machine. Specifically, some subset of the objects, $\mathbb{B} \subset \mathbb{X}$, are said to be "blickets" in that they have a causal influence on whether the blicket machine turns on. Thus, the causal graph in this scenario always take the form of a common effect, with edges $\{X_i \to M : X_i \in \mathbb{B}\}$.

The conjunctive and disjunctive overhypotheses specify the form of the blickets' causal influence on the machine. For example, let $X_i, X_j \in \mathbb{B}$ be blickets. In the conjunctive case, both objects are needed at the same time to turn the machine on. Formally, $P(M=1|X_i=1, X_j=1)=1$, while $P(M=0|X_i=1)=1$ and $P(M=0|X_j=1)=1$. In the disjunctive case, only one object (either X_i or X_j) is needed to turn the machine on, so $P(M=1|X_i=1)=1$ and $P(M=1|X_j=1)=1$. An illustration of the disjunctive versus conjunctive overhypotheses can be found in Figure 1. We note that while this setup is deterministic, it can easily be adapted to a stochastic environment with minimal modifications.

Experiments with SOTA RL Models To adapt the virtual blicket environment for agent learning in this work, we made the following design choices regarding the observations, actions, reward, and termination conditions.

Observations: We could allow the algorithms to observe the same embodied visual space as the children, but this places RL algorithms at a significant disadvantage, since they would need to not only understand causal structures, but also learn visual inputs and object detection. Thus, we choose to evaluate the algorithms in a purely symbolic environment where the objects are represented by one-hot indices. Formally, the state space is a vector in $o \in \{0,1\}^{N+1}$ where N is the number of blickets. The index $o_i, 0 \le i < N$ is 1 if o_i is on the detector, and 0 otherwise. The index o_N is 1 if the detector is illuminated, and 0 otherwise.

Actions: In the experiments with children, actions consist of placing a blicket onto the detector, removing a blicket from the detector, and pressing the "check" button to evaluate the detector's state. For RL algorithms, we simplify this process by allowing the agent to place multiple objects simultaneously, automatically "checking" the state of the detector with every action, and automatically resetting the detector after each check, leading to 2^N actions for N blickets. This means that the agent gets feedback with every action, which significantly improved training stability.

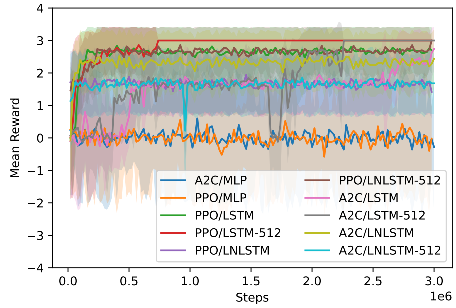
Reward: The reward function should capture whether the algorithm has learned the causal overhypothesis of an environment. To do this, we evaluate the models using a quiz-based framework. Models are allowed to make as many exploration steps as needed, and then trigger an action which switches to the evaluation mode. In the evaluation mode, models receive as input a blicket and must produce an action indicating if the object is a blicket or not. They receive a reward of 1 for identifying a correct blicket, and a reward of -1 for incorrectly labeling an object (i.e., for both false positives and false negatives). We also explored several reward modifications. In one modification, models were asked to disambiguate between Disjunctive and Conjunctive environments, with +1 reward for identifying the correct modality. The environment is implemented using the standard OpenAI Gym (Brockman et al., 2016) interface, allowing it to be used across many different pre-existing machine learning architectures and algorithms.

Deep Reinforcement Learning Algorithms

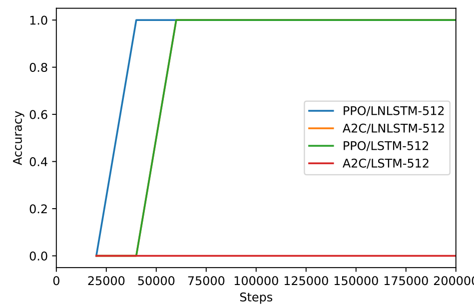
We evaluate the performance of two popular deep reinforcement learning algorithms, Advantage Actor Criric (A2C) (Mnih et al., 2016) and Proximal Policy Optimization (PPO2) (Schulman et al., 2017), on the blicket environment. For each algorithm, we use several policy variants: a standard MLP policy (no memory), an LSTM-based policy, and an LSTM-based policy with layer normalization with two hidden dimension variants of 256 and 512. For all of these policies, we found that a network with a hidden size of 512 obtained the optimal performance See the Appendix for learning hyperparameters. We train all of these algorithms in the *given hypothesis* scenario, where the agent is exposed during training to all possible overhypotheses, and asked to perform well given these scenarios.

Experimental Design We terminate training either after 3 million environment steps or when the agent obtains maximum reward for 500 consecutive episodes, whichever comes first. Each episode has 25 timesteps, so each agent is exposed to at most 120,000 episodes. To evaluate whether the agent can generalize to additional causal situations, we also train agents on five held-out scenarios: holding out all of the conjunctive overhypotheses, holding out all of the disjunctive overhypotheses, and holding out either one conjunctive overhypothesis or one disjunctive overhypothesis. The maximum achievable reward for these experiments is 3.

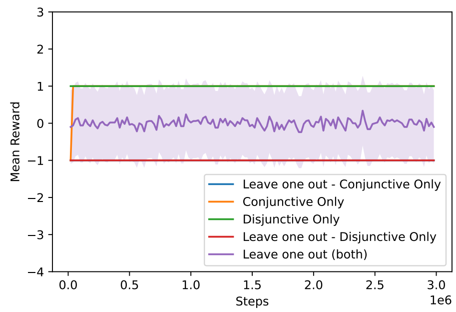
Results Figure 2a shows the performance of A2C and PPO2 when none of the hypotheses are held out. PPO2 outperforms A2C in almost all scenarios, achieving higher rewards faster. Further, the LSTM models clearly outperform the non



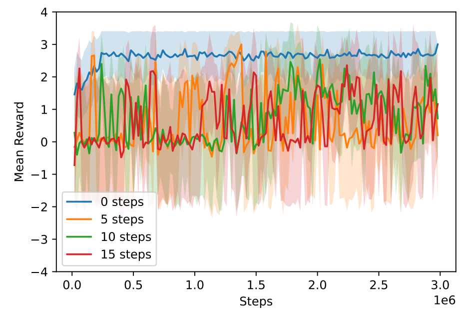
(a) Performance of PPO2 and A2C algorithms with MLP, LSTM, and layer-norm LSTM policies on the blicket environment.



(b) Accuracy of models over time when determining if the environment is conjunctive or disjunctive.



(c) Performance on several over-hypothesis variants. Conjunctive Only: Train on conjunctive hypotheses, test on conjunctive and disjunctive. Disjunctive Only: Train on only disjunctive hypotheses, test on conjunctive and disjunctive. Leave one out modifier: Train on 2/3 (or 4/6 for both) of the hypothesis within the overhypothesis, and test on the remaining.



(d) Performance of a standard A2C-LSTM model when forced to explore for K steps before entering the quiz phase. While this exploration helps the decision transformer, additional forced explanation is harmful to all standard RL models.

Figure 2: Experiments with standard deep reinforcement learning algorithms

memory-based models; this is expected, since causal learning requires memory. Unfortunately, none of the algorithms perform well on held-out causal examples, as shown in Figure 2c. This suggests that they are primarily learning to memorize the causal patterns, and thus are incapable of generalization. This conclusion comes with a caveat: because there are only a handful of possible hypotheses, it may be possible that we do not have enough data to perform well on held-out samples. Agents are, however, able to easily distinguish between conjunctive and disjunctive environments—Figure 2b shows that after very few steps, in the held-out situation (with both overhypotheses), agents can distinguish conjunctive from disjunctive environments, even though they are unable to determine which objects exactly are blickets.

Behavior Cloning

Recently Chen et al. (2021) introduced the Decision Transformer, a simple transformer-based approach to imitation learning, shown to outperform most existing behavior cloning methods. The decision transformer works by applying a causally-masked transformer to predict the reward-to-go of a flattened sequence of (state, action, reward, next state) tuples, with an ℓ^2-norm loss. The model then chooses actions during test time that maximize the predicted reward-to-go.

Model and Pre-Training Dataset	Reward	FCA
Decision Transformer		
Random	0.16 ± 1.747	0.67
Random (5-step forced exploration)	2.06 ± 0.998	0.71
Random (10-step forced exploration)	2.16 ± 0.872	0.73
Random (15-step forced exploration)	2.37 ± 0.633	0.77
A2C-LSTM	0.14 ± 1.847	0.63
PPO2-LSTM	0.26 ± 0.990	0.66
A2C-LayerNorm-LSTM	0.18 ± 0.983	0.62
PPO2-LayerNorm-LSTM	0.74 ± 1.460	0.64
Behavior Cloning		
Random	0.22 ± 0.975	0.61
Random (5-step forced exploration)	-0.16 ± 0.987	0.58
Random (10-step forced exploration)	0.14 ± 0.990	0.60
Random (15-step forced exploration)	0.04 ± 0.999	0.61
A2C-LSTM	0.10 ± 0.995	0.55
PPO2-LSTM	0.06 ± 0.998	0.63
A2C-LayerNorm-LSTM	0.04 ± 2.087	0.59
PPO2-LayerNorm-LSTM	0.12 ± 0.995	0.68

Table 1: Performance of the imitation learning models with different pre-training datasets. Reward is on the blicket-quiz task: +1 for correctly identifying a blicket, and -1 for incorrectly identifying a blicket. FCA refers to Forced Choice Accuracy, accuracy of the model in determining if the environment is conjunctive or disjunctive.

Experimental Design We collect trajectories for behavior cloning by exploring randomly in the space (an approach followed by Chen et al. (2021)), as well as with standard behavior cloning. To do so, we adapt the approach from Chen et al. (2021) to predict discrete actions in our space by adding a sigmoid activation to the action predictions and altering the action prediction loss accordingly. We evaluate the model using a target reward of 3, corresponding to identifying all of the blickets correctly, with additional training hyper-parameters given in the Appendix.

Results Table 1 shows the performance of the decision transformer and behavior cloning models when applied to datasets generated by several policies. As we can see, while expert policies allow for higher rewards using the decision transformer, allowing the model additional forced exploration time is the most important factor. This suggests that on their own, the A2C and PPO2 trained policies do not lead to sufficient exploration for learning a strong model of the reward, whereas forcing additional exploration (even if it is random) is much more useful. Notably, standard behavior cloning performs very poorly, as copying actions with the same local observations under different overhypotheses will likely lead to incorrect or uninformative actions.

When training the decision transformer model, we found that in some cases, random (and even the PPO2/A2C trained) models were unable to explore efficiently, as they entered the quiz environment too soon. Note that random exploration will enter the quiz environment after t steps with with probability $p = 1 - \frac{1}{2^t}$. Thus, we found it helpful to force the policy to explore for several steps before allowing it to enter the quiz environment. While this process helps the decision transformer, it negatively affects the performance of the standard RL models, as shown in Figure 2d.

Large Language Models

Recently large language models (Brown et al., 2020; Chowdhery et al., 2022) trained autoregressively on a large corpus of text to predict the next token when given a sequence of tokens have shown promising performance on a wide variety of tasks, including logical inference, common-sense reasoning and even causal reasoning (Veitch, Sridhar, & Blei, 2020; X. Wang, Xu, Tong, Roberts, & Liu, 2021), however have not been applied to causal overhypothesis tasks. Motivated by these approaches, we investigate the performance of two such language models, GPT-3 (Brown et al., 2020) and PaLM (Chowdhery et al., 2022), on the blicket environment using a purely text-based interaction with the model in which we provide a prompt to the model and evaluate its output. This prompt could be *freeform*, in which it has no set structure, or it could be *few-shot*, where it contains a small number of examples of correct prompt-output pairs, followed by the test prompt.

As in the experiments with children, there are four conditions: *given* versus *not given hypotheses*, and *disjunctive* versus *conjunctive*. In the *disjunctive* case, one of the three objects is a blicket. In the *conjunctive* case, two of the three objects are blickets. We evaluate how well the models can: 1) identify the blickets and 2) identify whether the causal structure is disjunctive or conjuctive. For GPT-3, we use the OpenAI API[1].

Condition, Model, and Input	Blickets Chosen	Causal Structure
Given hypotheses, disjunctive		
GPT-3, freeform	1/1 correct, 6 wrong	**correct**
PaLM, freeform	1/1 correct, 1 wrong	wrong
PaLM, two-shot	**1/1 correct**	**correct**
Given hypotheses, conjunctive		
GPT-3, freeform	2/2 correct, 1 wrong	**correct**
PaLM, freeform	**2/2 correct**	**correct**
PaLM, two-shot	**2/2 correct**	**correct**
Not given hypotheses, disjunctive		
GPT-3, freeform	1/1 correct, 7 wrong	—
PaLM, freeform	0/1 correct	—
PaLM, two-shot	1/2 correct, 1 wrong	—
Not given hypotheses, conjunctive		
GPT-3, freeform	2/2 correct, 7 wrong	—
PaLM, freeform	**2/2 correct**	—
PaLM, two-shot	1/2 correct, 1 wrong	—

Table 2: Results for GPT-3 and PaLM for the four conditions. Bold font indicates a fully correct answer. For the *not given hypotheses* setting, answering the causal structure question is impossible, as there is not enough information to determine whether the blicket detectors in the training phase are disjunctive or conjunctive. For GPT-3, we use the text-davinci-002 model, with temperature 0.7, maximum length 256, and frequency and presence penalties 0. For PaLM, we use a temperature of 0 to obtain the greedy, or one-best decoding. When given the freeform input, the PaLM model output continues indefinitely, so we take only the first two sentences of the output.

Freeform Prompt In the freeform prompt, we provide the model with text that is as similar as possible to what children receive in the blicket experiment. The only difference is that we replace the visual components with text descriptions. We include the exact prompts in the Appendix. The prompt starts with a description of interacting with a striped blicket machine and then a dotted blicket machine, each of which has three unique objects. There are three interaction examples given per machine, for example "If we put the blue pyramid on the machine, then it does not light up". In the *given hypotheses* condition, the striped blicket machine is conjunctive and the dotted blicket machine is disjunctive. In the *not given hypotheses* condition, there is not enough information to determine whether the striped and dotted blicket machines are disjunctive or conjuctive. Thus the *given hypotheses* condition defines the space of overhypotheses, whereas the *not given hypotheses* condition does not. The prompt then introduces a new blicket machine, along with examples of interactions with the machine and whether it lights up or not. Finally we ask the model which objects are blickets and whether the new machine is

[1]Available at `https://beta.openai.com/overview`

more similar to the striped or dotted machine.

Few-shot Prompt In the few-shot prompt, we structure the input into two prompt-output demonstrations, one per machine, containing the same information as in the freeform prompt. Unlike the information given to children, in the outputs we explicitly state which objects are blickets. In addition, for the *given hypotheses* condition, in the preamble we define the disjunctive and conjunctive hypotheses: "A striped machine needs two blickets to make it light up, and a dotted machine needs one blicket to make it light up", and in the outputs we state whether this is more similar to a striped or dotted machine.

The results are reported in Table 2. We find that when given hypotheses, GPT-3 and PaLM are almost always able to select the correct causal structure, but they are not always able to select the correct blickets. In particular, GPT-3 frequently names too many objects as blickets, including those associated with other machines. In contrast, PaLM never identified objects as blickets that were not one of the three objects in the test task. In the two-shot setting, PaLM performs best when the space of overhypotheses is covered perfectly by the two examples given, as one would expect. However, when this space is not covered, i.e. in the *not given hypothesis* setting, PaLM struggles in the two-shot setting because the test example's causal structure does not match either of the two given examples. Adding chain-of-thought reasoning in the PaLM prompt did not improve results in either setting.

Q-Learning

We also train tabular Q-Learning on the symbolic version of the blicket environment. We append the full history of previous observations to the current observation in order to give the agent memory. The Q-values are initialized to zero. We used ε-greedy exploration with an exploration probability of 0.1, and we found the best learning rate was 0.95. Q-learning is able to learn the task very quickly due to a small search space—it took an average of 70 episodes and 292 steps for the agent to converge to maximum reward. However, tabular Q-learning is incapable of generalizing to new scenarios, so we do not test Q-Learning agents in the held-out scenarios.

Causal Discovery Baselines

Most deep learning models do not explicitly model the causal structure of the data. Thus, to explore the performance of existing causal discovery models on the benchmark, we used the recent Differentiable Causal Discovery from Interventional Data (DCDI) approach (Brouillard, Lachapelle, Lacoste, Lacoste-Julien, & Drouin, 2020). We evaluated the model on causal structure learning for both disjunctive and conjunctive settings. For disjunctive settings, the model achieved a structured hamming distance (SHD) of 2.3. Noting that the graph has 4 variables and hence 16 possible edges in the graph structure, the model achieved a SHD ratio of 85%. For the conjunctive setting, the model achieved a SHD of 2.6, a ratio of 83%.

Discussion & Conclusion

In this work, we looked into evaluating and understanding how machine learning models learn causal overhypotheses, by evaluating these models in the blicket environment. In contrast to existing benchmark tasks, in which there is a fixed causal structure, this environment focuses on the need for *causal overhypotheses* in order to explore effectively to determine the underlying causal structure. We focused on three categories of state-of-the-art methods—deep RL, behavior cloning, and large language models—for tasks in this environment. In blicket detector experiments, children are able to learn causal overhypotheses from only a handful of observations and can apply these overhypotheses to explore effectively for a new situation (Kosoy et al., 2022). In contrast, our experiments indicate that state-of-the-art machine learning algorithms have difficulty learning and using causal overhypotheses for exploration and inference—we saw this in the weak performance of deep RL algorithms on held-out environments and in the tendency of decision transformer models to under-explore. With language models, we provide the same observations that the children were given in Kosoy et al. (2022), along with a full set of examples for the new situation (thus removing the need for exploration). Despite this, language models struggle when the hypotheses are not given, and are not able to express uncertainty about the causal structure in that case.

Given that understanding and leveraging causal structure is essential to developing general intelligence, this work highlights an opportunity for improvement in this direction, and provides a set of concrete benchmark tasks to measure improvement. One direction of future work is to build machine learning models that can better learning causal overhypotheses. Modular architectures have shown to be helpful in understanding causal hypotheses of the environment (Goyal et al., 2019, 2021; Ke et al., 2021); it would be promising to explore such models for causal overhypotheses understanding. Another direction of future work is to improve on exploration in RL agents by explicitly learning and incorporating causal overhypotheses, in order to narrow down the search over possibilities. Furthermore, an interesting direction is to train models on children's exploration behavior used in solving causal problems, for instance the trajectories provided by Kosoy et al. (2020).

In the real world, there exist many types of overhpotheses, as well as conditional probability distributions and it remains essential future work to extend either the blicket or other RL environments (such as (Ahmed et al., 2020; Ke et al., 2021)) to include other types of causal overhypotheses. Indeed, with minor changes, one could extend the blicket environment to test other kinds of causal overhypotheses, such as inferring whether systems are stochastic or deterministic, or require sequential or unordered interventions. This environment also allows for multiple measures of causal inference, including interventions and counterfactuals, as well as predictions.

References

Ahmed, O., Träuble, F., Goyal, A., Neitz, A., Bengio, Y., Schölkopf, B., ... Bauer, S. (2020). Causalworld: A robotic manipulation benchmark for causal structure and transfer learning. *arXiv preprint arXiv:2010.04296*.

Allen, K. R., Smith, K. A., & Tenenbaum, J. B. (2020). Rapid trial-and-error learning with simulation supports flexible tool use and physical reasoning. *Proceedings of the National Academy of Sciences*, *117*(47), 29302–29310. Retrieved from `https://www.pnas.org/content/117/47/29302` doi: 10.1073/pnas.1912341117

Bakhtin, A., van der Maaten, L., Johnson, J., Gustafson, L., & Girshick, R. (2019). Phyre: A new benchmark for physical reasoning. In *Advances in neural information processing systems* (pp. 5082–5093).

Bengio, Y., Deleu, T., Rahaman, N., Ke, R., Lachapelle, S., Bilaniuk, O., ... Pal, C. (2019). A meta-transfer objective for learning to disentangle causal mechanisms. *arXiv preprint arXiv:1901.10912*.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., & Zaremba, W. (2016). *Openai gym.*

Brouillard, P., Lachapelle, S., Lacoste, A., Lacoste-Julien, S., & Drouin, A. (2020). Differentiable causal discovery from interventional data. *Advances in Neural Information Processing Systems*, *33*, 21865–21877.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... others (2020). Language models are few-shot learners. *Advances in neural information processing systems*, *33*, 1877–1901.

Chen, L., Lu, K., Rajeswaran, A., Lee, K., Grover, A., Laskin, M., ... Mordatch, I. (2021). Decision transformer: Reinforcement learning via sequence modeling. *arXiv preprint arXiv:2106.01345*.

Chevalier-Boisvert, M., Bahdanau, D., Lahlou, S., Willems, L., Saharia, C., Nguyen, T. H., & Bengio, Y. (2018). Babyai: A platform to study the sample efficiency of grounded language learning. In *International conference on learning representations.*

Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., ... others (2022). Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Cobbe, K., Klimov, O., Hesse, C., Kim, T., & Schulman, J. (2018). Quantifying generalization in reinforcement learning. *arXiv preprint arXiv:1812.02341*.

Gopnik, A. (2012). Scientific thinking in young children: Theoretical advances, empirical research, and policy implications. *Science*, *337*(6102), 1623–1627.

Gopnik, A., O'Grady, S., Lucas, C. G., Griffiths, T. L., Wente, A., Bridgers, S., ... Dahl, R. E. (2017). Changes in cognitive flexibility and hypothesis search across human life history from childhood to adolescence to adulthood. *Proceedings of the National Academy of Sciences*, *114*(30), 7892–7899.

Gopnik, A., & Sobel, D. M. (2000). Detecting blickets: how young children use information about novel causal powers in categorization and induction. *Child Development*, *71*(5), 1205–1222.

Gopnik, A., & Wellman, H. M. (2012). Reconstructing constructivism: Causal models, bayesian learning mechanisms, and the theory theory. *Psychological bulletin*, *138*(6), 1085.

Goyal, A., Didolkar, A., Ke, N. R., Blundell, C., Beaudoin, P., Heess, N., ... Bengio, Y. (2021). Neural production systems. *Advances in Neural Information Processing Systems*, *34*, 25673–25687.

Goyal, A., Lamb, A., Hoffmann, J., Sodhani, S., Levine, S., Bengio, Y., & Schölkopf, B. (2019). Recurrent independent mechanisms. *arXiv preprint arXiv:1909.10893*.

Griffiths, T. L., & Tenenbaum, J. B. (2009). Theory-based causal induction. *Psychological review*, *116*(4), 661.

Hill, A., Raffin, A., Ernestus, M., Gleave, A., Kanervisto, A., Traore, R., ... Wu, Y. (2018). *Stable baselines.* `https://github.com/hill-a/stable-baselines`. GitHub.

James, S., Ma, Z., Arrojo, D. R., & Davison, A. J. (2020). Rlbench: The robot learning benchmark & learning environment. *IEEE Robotics and Automation Letters*, *5*(2), 3019–3026.

Ke, N. R., Didolkar, A. R., Mittal, S., Goyal, A., Lajoie, G., Bauer, S., ... Pal, C. (2021). Systematic evaluation of causal discovery in visual model based reinforcement learning. *arXiv preprint arXiv:2107.00848*.

Kemp, C., Goodman, N. D., & Tenenbaum, J. B. (2010). Learning to learn causal models. *Cognitive Science*, *34*(7), 1185–1243.

Kemp, C., Perfors, A., & Tenenbaum, J. B. (2007). Learning overhypotheses with hierarchical bayesian models. *Developmental science*, *10*(3), 307–321.

Kosoy, E., Collins, J., Chan, D. M., Huang, S., Pathak, D., Agrawal, P., ... Hamrick, J. B. (2020). Exploring exploration: Comparing children with rl agents in unified environments. *arXiv preprint arXiv:2005.02880*.

Kosoy, E., Liu, A., Collins, J. L., Chan, D., Hamrick, J. B., Ke, N. R., ... Gopnik, A. (2022). Learning causal overhypotheses through exploration in children and computational models. In *First conference on causal learning and reasoning.* Retrieved from `https://openreview.net/forum?id=6GLEuG0d8i`

Lin, S., Hilton, J., & Evans, O. (2022). Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*.

Lucas, C. G., Bridgers, S., Griffiths, T. L., & Gopnik, A. (2014). When children are better (or at least more open-minded) learners than adults: Developmental differences in learning the forms of causal relationships. *Cognition*, *131*(2), 284–299.

Lucas, C. G., & Griffiths, T. L. (2010). Learning the form of causal relationships using hierarchical bayesian models. *Cognitive Science*, *34*(1), 113–147.

Machado, M. C., Bellemare, M. G., Talvitie, E., Veness, J., Hausknecht, M., & Bowling, M. (2018). Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. *Journal of Artificial Intelligence Research*, *61*, 523–562.

McDuff, D., Song, Y., Lee, J., Vineet, V., Vemprala, S., Gyde, N., ... Kapoor, A. (2021). Causalcity: Complex simulations with agency for causal discovery and reasoning. *arXiv preprint arXiv:2106.13364*.

Mnih, V., Badia, A. P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., ... Kavukcuoglu, K. (2016). Asynchronous methods for deep reinforcement learning. In *International conference on machine learning* (pp. 1928–1937).

Nichol, A., Pfau, V., Hesse, C., Klimov, O., & Schulman, J. (2018). Gotta learn fast: A new benchmark for generalization in rl. *arXiv preprint arXiv:1804.03720*.

Perfors, A., Tenenbaum, J. B., Griffiths, T. L., & Xu, F. (2011). A tutorial introduction to bayesian models of cognitive development. *Cognition*, *120*(3), 302–321.

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

Schölkopf, B., Locatello, F., Bauer, S., Ke, N. R., Kalchbrenner, N., Goyal, A., & Bengio, Y. (2021). Towards causal representation learning. *arXiv preprint arXiv:2102.11107*.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Sontakke, S. A., Mehrjou, A., Itti, L., & Schölkopf, B. (2021). Causal curiosity: Rl agents discovering self-supervised experiments for causal representation learning. In *International conference on machine learning* (pp. 9848–9858).

Tenenbaum, J. B., Kemp, C., Griffiths, T. L., & Goodman, N. D. (2011). How to grow a mind: Statistics, structure, and abstraction. *science*, *331*(6022), 1279–1285.

Veitch, V., Sridhar, D., & Blei, D. (2020). Adapting text embeddings for causal inference. In *Conference on uncertainty in artificial intelligence* (pp. 919–928).

Wang, J. X., King, M., Porcel, N., Kurth-Nelson, Z., Zhu, T., Deck, C., ... Botvinick, M. (2021). Alchemy: A structured task distribution for meta-reinforcement learning. *arXiv preprint arXiv:2102.02926*.

Wang, X., Xu, X., Tong, W., Roberts, R., & Liu, Z. (2021). Inferbert: a transformer-based causal inference framework for enhancing pharmacovigilance. *Frontiers in Artificial Intelligence*, *4*, 659622.

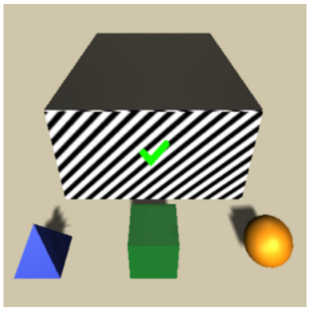
Watkins, C. J., & Dayan, P. (1992). Q-learning. *Machine learning*, *8*(3), 279–292.

Yu, T., Quillen, D., He, Z., Julian, R., Hausman, K., Finn, C., & Levine, S. (2019). Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. *arXiv preprint arXiv:1910.10897*.

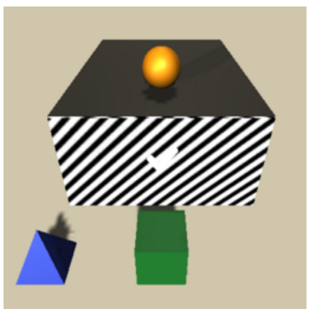
Zhang, C., Jia, B., Edmonds, M., Zhu, S.-C., & Zhu, Y. (2021). Acre: Abstract causal reasoning beyond covariation. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 10643–10653).

Appendix

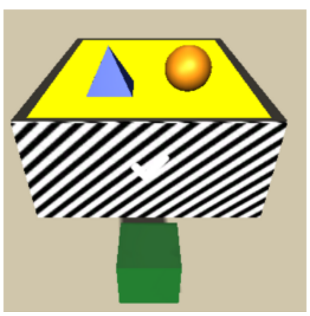
Additional Environment Details



(a) No objects on the detector



(b) One object on the detector, but detector does not light up.



(c) Two objects, triggering the detector

Figure 3: A visualization of some of the possible states of the blicket environment when rendered in Unity.

A2C and PPO2 Hyperparameters

Table 3 gives the set of hyper-parameters for the PPO2 and A2C algorithms used in the paper. We use the reference implementations from Stable Baselines (Hill et al., 2018) for our standard RL models. For model size variants, we use the standard policy networks available in Hill et al. (2018) with no additional modification (asides from the hidden dimension, as specified in the main paper).

Behavior Cloning Hyperparameters

The decision transformer was trained using a batch size of 128, a K-value of 30, embedding dimension of 128, 3 layers, one head, and dropout of 0.1. For standard behavior cloning, we use an MLP with a hidden-dimension of 128. The weights are optimized using ADAM, with a learning rate of $1e^{-4}$ and weight decay of $1e^{-4}$. Both models are trained for 10 epochs using one Nvidia Titan-X Maxwell GPU, which takes less than one hour, and the best validation set checkpoint is used for each during test time. Models are evaluated on 100 rollouts in the environment, and the mean reward is reported.

	A2C	PPO2
Discount Factor	0.99	0.99
Steps/Update	5	128
Value Function Coefficient	0.25	0.5
Entropy Coefficient	0.01	0.01
Learning Rate	0.0007	0.00025
LR Schedule	Constant	Constant
Gradient Clipping (Max Norm)	0.5	0.5
GAE Bias/Variance Lambda	-	0.95

Table 3: Hyperparameters for A2C and PPO2 RL algorithms.

Language Model Prompts

The prompts we used for GPT-3 and PaLM are based on the instructions and examples that children are given in (Kosoy et al., 2022).

Freeform Prompts

We modelled the freeform prompt as closely as possible to the experiment done with children in (Kosoy et al., 2022). The main difference is that we replace the visual components with textual descriptions. The freeform prompt first explains that blicket machines turn on when objects called blickets are placed on them. It explains that some objects are blickets and some are not, and machines need either one, two, or three blickets placed on it in order to turn on. Next, the prompt introduces a striped blicket machine and three objects, and gives three examples of whether the machine lights up or not when certain objects are placed on it—for example, "If we put the blue pyramid on the machine, then it does not light up". Then the prompt does the same for the dotted blicket machine and three different objects. Finally, the prompt does the same for a new blicket machine with three different objects, and asks which of these objects are blickets, and whether this new machine works like the striped machine or like the dotted machine.

We tested the language models in four conditions, each with its own prompt. The conditions consist of either hypothesis given or not given, combined with either disjunctive or conjunctive causal structure. The new machine either has a conjunctive or disjunctive causal structure, depending on what the condition is. In the conditions where the hypothesis is given, it is clear from the examples that the striped machine has a conjunctive causal structure and the dotted machine has a disjunctive structure. In the conditions where the hypothesis is not given, it is *not* clear from the examples whether the striped and dotted machines have conjunctive or disjunctive structure. Thus in the not-given hypothesis conditions, when the model is asked whether the new machine works like the striped or dotted machine, the correct response is to be unsure.

Below are the exact freeform prompts for all four conditions; the same prompts are given to both GPT-3 and PaLM.

- **All conditions:** A blicket detector is a special kind of machine, objects that are different colors and shapes either make the machine turn on or not. If the object is a blicket and placed on the machine then the machine will turn on. Sometimes 1, 2 or 3 blickets make the machine turn on. Our goal is to make the machine turn on and figure out which shapes make it do so.

  Can you tell me which objects are blickets? Does this checkerboard pattern blicket detector behave like the striped pattern blicket detector or like the dotted pattern blicket detector?

- **Given hypotheses:** First I have a striped pattern blicket detector, it behaves in the following way: I have 3 objects, one blue pyramid, one green cube and one orange sphere. First I put the blue pyramid on the striped pattern blicket machine and it does not light up. Then I put the orange sphere on the striped pattern blicket machine and it does not light up. Then I put the blue pyramid and the orange sphere on the striped pattern blicket machine and it did light up!

  Then I have a dotted pattern blicket detector. I have 3 different objects now, a yellow cylinder, a purple cone, and a red dome. First I put the purple cone on the dotted pattern blicket detector and it did light up! Then I put the yellow cylinder on the dotted pattern blicket detector and it does not light up. Then I put the yellow cylinder and the purple cone on the dotted pattern blicket detector and it did light up!

- **Not-given hypotheses:** First I have a striped pattern blicket detector, it behaves in the following way: I have 3 objects, one blue pyramid, one green cube and one orange sphere. First I put the blue pyramid on the striped pattern blicket detector and it does not light up. Then I put the green cube on the striped pattern blicket detector and it does not light up. Then I put the blue pyramid and the orange sphere on the striped pattern blicket detector and it did light up!

  Then I have a dotted pattern blicket detector. I have 3 different objects now, a yellow cylinder, a purple cone, and a red dome. First I put the purple cone on the dotted pattern blicket detector and it does not light up. Then I put the yellow cylinder on the dotted pattern blicket detector and it does not light up. Then I put the red half dome and the purple cone on the dotted pattern blicket detector and it did light up!

- **Disjunctive:** Then I have a checkerboard pattern blicket detector. I have 3 new objects, a teal prism, a pink frustum and a brown torus. This machine could work like the dotted patterned blicket detector or it could work like the striped pattern blicket detector. First I put the brown torus on the checkerboard pattern blicket detector and it does light up! Then I put the pink frustum on the checkerboard pattern blicket detector and it does not light up. Then I put the teal prism on the checkerboard pattern blicket detector and it does not light up. Then I put the brown torus and the pink frustum on the checkerboard pattern blicket detector and it did light up! Then I put the teal prism and the brown torus on the checkerboard pattern blicket detector and it did light up! Then I put the teal prism and the pink frustum and the brown torus on the checkerboard pattern blicket detector and it did light up! Then I put the pink frustum and the teal prism on the checkerboard pattern blicket detector and it does not light up.

- **Conjunctive:** Then I have a checkerboard pattern blicket detector. I have 3 new objects, a teal prism, a pink frustum and a brown torus. This machine could work like the dotted patterned blicket detector or it could work like the striped pattern blicket detector. First I put the brown torus on the checkerboard pattern blicket detector and it does not light up. Then I put the pink frustum and the brown torus on the checkerboard pattern blicket detector and it does not light up. Then I put the teal prism on the checkerboard pattern blicket detector and it does not light up. Then I put the teal prism and the pink frustum on the checkerboard pattern blicket detector and it does not light up. Then I put the pink frustum on the checkerboard pattern blicket detector and it does not light up. Then I put the teal prism and the brown torus on the checkerboard pattern blicket detector and it does light up! Then I put the teal prism and the pink frustum and the brown torus on the checkerboard pattern blicket detector and it did light up!

Below are the outputs that GPT-3 produces for each of the conditions, at three differnet settings for temperature, 0, 1 and 0.7:

- **Temperature = 0.7, Given hypothesis, disjunctive:** The objects that are blickets are the blue pyramid, the orange sphere, the purple cone, the yellow cylinder, the brown torus, the pink frustum and the teal prism. This checkerboard pattern blicket detector behaves like the dotted pattern blicket detector.
- **Temperature = 0.7, Given hypothesis, conjunctive:** The objects that are blickets are the teal prism, the pink frustum and the brown torus. This checkerboard pattern blicket detector behaves like the striped pattern blicket detector.
- **Temperature = 0.7, Not-given hypothesis, disjunctive:** The objects that are blickets are the blue pyramid, the orange sphere, the yellow cylinder, the purple cone, the red

dome, the teal prism, the pink frustum, and the brown torus. This checkerboard pattern blicket detector behaves like the striped pattern blicket detector.

- **Temperature = 0.7, Not-given hypothesis, conjunctive:** The objects that are blickets are the blue pyramid, the green cube, the orange sphere, the yellow cylinder, the purple cone, the red dome, the teal prism, the pink frustum, and the brown torus. This checkerboard pattern blicket detector behaves like the dotted pattern blicket detector.

- **Temperature = 0, Given hypothesis, disjunctive:** The objects that are blickets are the blue pyramid, the orange sphere, the purple cone, the yellow cylinder, the brown torus, the pink frustum and the teal prism. The checkerboard pattern blicket detector behaves like the striped pattern blicket detector.
- **Temperature = 0, Given hypothesis, conjunctive:** The objects that are blickets are the blue pyramid, the orange sphere, the purple cone, the yellow cylinder, the teal prism, the pink frustum and the brown torus. The checkerboard pattern blicket detector behaves like the striped pattern blicket detector.
- **Temperature = 0, Not-given hypothesis, disjunctive:** The objects that are blickets are the blue pyramid, the orange sphere, the purple cone, the yellow cylinder, the brown torus, the pink frustum and the teal prism. The checkerboard pattern blicket detector behaves like the striped pattern blicket detector.
- **Temperature = 0, Not-given hypothesis, conjunctive:** The objects that are blickets are the blue pyramid, the orange sphere, the yellow cylinder, the purple cone, the red dome, the teal prism, the pink frustum, and the brown torus. This checkerboard pattern blicket detector behaves like the striped pattern blicket detector.

- **Temperature = 1, Given hypothesis, disjunctive:** The checkerboard pattern blicket detector behaves like the dotted pattern blicket detector.
- **Temperature = 1, Given hypothesis, conjunctive:** From the data, it seems that the checkerboard pattern blicket detector behaves like the dotted pattern blicket detector. This is because in order for the machine to turn on, two blickets are needed.
- **Temperature = 1, Not-given hypothesis, disjunctive:** The objects that are blickets are the blue pyramid, the orange sphere, the purple cone, the red dome, the brown torus, the pink frustum, and the teal prism. This checkerboard pattern blicket detector behaves like the striped pattern blicket detector.
- **Temperature = 1, Not-given hypothesis, conjunctive:** The objects that are blickets are the blue pyramid, the orange sphere, the purple cone, the yellow cylinder, the teal prism, the pink frustum and the brown torus. The checkerboard pattern blicket detector behaves like the striped pattern blicket detector.

Below are the outputs that PaLM produces for each of the conditions. We only take into account the first two sentences of the output when evaluating the model's performance.

- **Given hypothesis, disjunctive:** The answer is that the checkerboard pattern blicket detector behaves like the striped pattern blicket detector. The blickets are the brown torus and the pink frustum. The reason why the checkerboard pattern blicket detector behaves like the striped pattern blicket detector is because the machine only turns on when the brown torus and the pink frustum are on the machine. The machine does not turn on when the brown torus and the teal prism are on the machine. The machine does not turn on when the pink frustum and the teal prism are on the machine. The machine does not turn on...
- **Given hypothesis, conjunctive:** The checkerboard pattern blicket detector behaves like the striped pattern blicket detector. The blickets are the teal prism and the brown torus. The checkerboard pattern blicket detector behaves like the dotted pattern blicket detector. The blickets are the teal prism and the brown torus. The checkerboard pattern blicket detector behaves like the striped pattern blicket detector. The blickets are the teal prism and the pink frustum. The checkerboard pattern blicket detector behaves like the dotted pattern blicket detector. The blickets are the...
- **Not-given hypothesis, disjunctive:** This is a very difficult problem. It is not possible to figure out which objects are blickets and which are not. It is not possible to figure out how the checkerboard pattern blicket detector works. This is a very difficult problem. It is not possible to figure out which objects are blickets and which are not. It is not possible to figure out how the checkerboard pattern blicket detector works. This is a very difficult problem. It is not possible to figure out which objects are blickets and which are not. It is not possible to figure out how the checkerboard...
- **Not-given hypothesis, conjunctive:** The checkerboard pattern blicket detector behaves like the striped pattern blicket detector. The blickets are the teal prism and the brown torus. The checkerboard pattern blicket detector behaves like the dotted pattern blicket detector. The blickets are the teal prism and the brown torus. The checkerboard pattern blicket detector behaves like the striped pattern blicket detector. The blickets are the teal prism and the pink frustum. The checkerboard pattern blicket detector behaves like the dotted pattern blicket detector. The blickets are the...

Few-Shot Prompts

The few-shot prompts consist of a preface, two input-output examples, and a new input (that the model is must provide the corresponding output for). The input describes the three objects and three examples of whether the machine lights up or not when certain combinations of objects are placed on it. The output specifies which of the three objects are blickets, and whether the machine is like the striped or dotted machine.

Below are the exact texts of the few-shot prompts that we used as inputs to the PaLM model.

- **All conditions:** If we put objects on the machine, then it will either light up or not. Some objects are blickets, and others are not. A striped machine needs two blickets to make it light up, and a dotted machine needs one blicket to make it light up. Please identify which objects are blickets and whether the machine behaves like the striped machine or the dotted machine.

  <insert given/not-given hypothesis portion>

  <insert disjunctive/conjunctive portion>

- **Given hypotheses:**

  Input: There is a blue pyramid, a green cube, and an orange sphere. If we put the blue pyramid on the machine, then it does not light up. If we put the orange sphere on the machine, then it does not light up. If we put the blue pyramid and the orange sphere on the machine, then it does light up.

  Output: The blue pyramid and orange sphere are blickets. This machine behaves like the striped machine.

  Input: There is a yellow cylinder, a purple cone, and a red dome. If we put the purple cone on the machine, then it does light up. If we put the yellow cylinder on the machine, then it does not light up. If we put the yellow cylinder and the purple cone on the machine, then it does light up.

  Output: The purple cone is a blicket. This machine behaves like the dotted machine.

- **Not-given hypotheses:**

  Input: There is a blue pyramid, a green cube, and an orange sphere. If we put the blue pyramid on the machine, then it does not light up. If we put the green cube on the machine, then it does not light up. If we put the blue pyramid and the orange sphere on the machine, then it does light up.

  Output: The orange sphere is a blicket, and the blue pyramid is maybe a blicket.

  Input: There is a yellow cylinder, a purple cone, and a red dome. If we put the purple cone on the machine, then it does light up. If we put the yellow cylinder on the machine, then it does not light up. If we put the red dome and the purple cone on the machine, then it does light up.

  Output: The red dome is a blicket and the purple cone is maybe a blicket.

- **Disjunctive:**

  There is a teal prism, a pink frustum, and a brown torus. If we put the brown torus on the machine, then it does light up. If we put the pink frustum and the brown torus on the machine, then it does light up. If we put the teal prism on the machine, then it does not light up. If we put the teal prism and the pink frustum on the machine, then it does not light up. If we put the pink frustum on the machine, then it does not light up. If we put the teal prism and the brown torus on the machine, then it does light up. If we put the teal prism and the pink frustum and the brown torus on the machine, then it does light up.

  Output:

- **Conjunctive:**

  Input: There is a teal prism, a pink frustum, and a brown torus. If we put the brown torus on the machine, then it does not light up. If we put the pink frustum and the brown torus on the machine, then it does not light up. If we put the teal prism on the machine, then it does not light up. If we put the teal prism and the pink frustum on the machine, then it does not light up. If we put the pink frustum on the machine, then it does not light up. If we put the teal prism and the brown torus on the machine, then it does light up. If we put the teal prism and the pink frustum and the brown torus on the machine, then it does light up.

  Output:

Below are the outputs that PaLM produces for each of the conditions:

- **Given hypothesis, disjunctive:** The brown torus is a blicket. This machine behaves like the dotted machine.
- **Given hypothesis, conjunctive:** The teal prism and brown torus are blickets. This machine behaves like the striped machine.
- **Not-given hypothesis, disjunctive:** The brown torus is a blicket, and the pink frustum is maybe a blicket.
- **Not-given hypothesis, conjunctive:** The teal prism is a blicket, and the pink frustum is maybe a blicket.