A Appendix

A.1 Contrastive cross-entropy approximates generative log-likelihood

Consider a batch of images $\{\mathbf{x}_b\}_{b=1}^B$ and a caption $\mathbf{y}$ corresponding to one of the images $\mathbf{x}_j$. This caption will come either from the paired dataset or as a sample from the image-conditional language encoder $q_\omega(\mathbf{y} \mid \mathbf{x}_j)$.

The posterior distribution over the classification of the image index c is

$$\Pr(c = j \mid \{\mathbf{x}_b\}_{b=1}^B, \mathbf{y}) = \frac{p(\mathbf{x}_j \mid \mathbf{y}) \prod_{i \neq j} p(\mathbf{x}_i)}{\sum_b p(\mathbf{x}_b \mid \mathbf{y}) \prod_{k \neq b} p(\mathbf{x}_k)} = \frac{p(\mathbf{x}_j \mid \mathbf{y})/p(\mathbf{x}_j)}{\sum_b p(\mathbf{x}_b \mid \mathbf{y})/p(\mathbf{x}_b)}. \tag{8}$$

Therefore, the multi-class cross-entropy over the correct index is

$$L(\{\mathbf{x}_b\}_{b=1}^B, \mathbf{y}, c = j) = \log p(\mathbf{x}_j \mid \mathbf{y}) - \log p(\mathbf{x}_j) - \log \sum_b \frac{p(\mathbf{x}_b \mid \mathbf{y})}{p(\mathbf{x}_b)}. \tag{9}$$

We can manipulate the third term into the form of an expectation

$$L(\{\mathbf{x}_b\}_{b=1}^B, \mathbf{y}, c = j) = \log p(\mathbf{x}_j \mid \mathbf{y}) - \log p(\mathbf{x}_j) - \log \frac{1}{B} \sum_b \frac{p(\mathbf{x}_b \mid \mathbf{y})}{p(\mathbf{x}_b)} - \log B, \tag{10}$$

and for large B, we can approximate $\frac{1}{B} \sum_b \frac{p(\mathbf{x}_b|\mathbf{y})}{p(\mathbf{x}_b)} \approx \mathbb{E}_{p(\mathbf{x})}\left[\frac{p(\mathbf{x}|\mathbf{y})}{p(\mathbf{x})}\right] = \int_{\mathbf{x}} p(\mathbf{x} \mid \mathbf{y}) = 1$. In this large batch limit, the multi-class cross-entropy is proportional to the language-conditional image decoder log-likelihood up to constant factors in $\mathbf{y}$:

$$L(\{\mathbf{x}_b\}_{b=1}^B, \mathbf{y}, c = j) \approx \log p(\mathbf{x}_j \mid \mathbf{y}) + \text{constant}(\mathbf{y}). \tag{11}$$

Therefore, if we have a multi-class classifier that discriminates if a batch element $\mathbf{y}$ is paired to a batch element $\mathbf{x}$, we can substitute the generative log likelihood with the classifier's loss:

$$\mathbb{E}_{q_\omega(\mathbf{y}|\mathbf{x}_j)}[\log p_\theta(\mathbf{x}_j \mid \mathbf{y})] = \mathbb{E}_{q_\omega(\mathbf{y}|\mathbf{x}_j)}[L(\{\mathbf{x}_b\}_{b=1}^B, \mathbf{y}, c = j)] + \text{constant}(\mathbf{y}), \tag{12}$$

and gradients with respect to $q_\omega(\mathbf{y} \mid \mathbf{x}_j)$ are the same in expectation.

A.2 Details for caption data collection

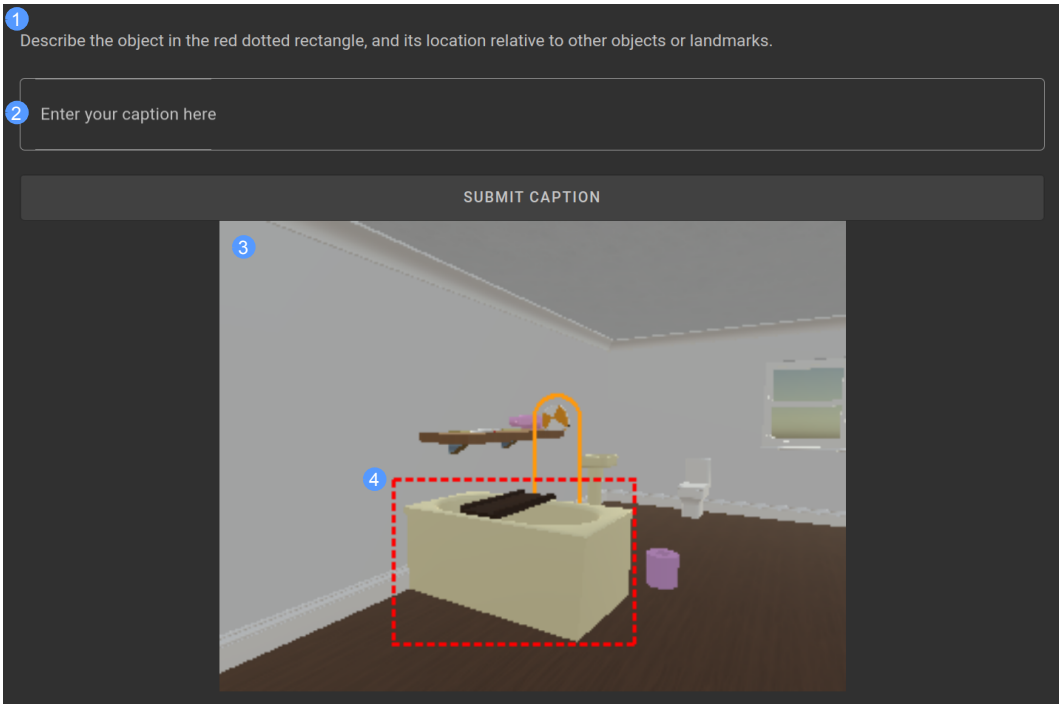

Figure A1: User interface and instructions for caption data collection. We recruited participants (N = 80) through an internal crowdsourcing pool, and collected a total of 78K captions using a total of $4,000$ participant hours. The full details of our study design, including compensation rates, were reviewed by our institution's independent ethical review committee. All participants provided informed consent prior to completing tasks and were reimbursed for their time. Participants were provided with a link to the caption interface and the following task description:

> You will be shown an image of a room, where one of the objects is highlighted with a **dotted red rectangle**. Follow the script, think of a description of the **object and its surroundings** and type it in the text field. The description should also have enough details such that another player in the room can easily find and understand which object you are referring to.

During data collection, the caption interface displayed a single frame randomly selected from human-human interaction data as described by [22] at a resolution of 320×240. An object from the image was randomly selected and highlighted by a bounding box. The participants were prompted to "[d]escribe the object in the red dotted rectangle, and its location relative to other objects or landmarks". After the participants input the caption in the text-box and clicked "submit", the image is refreshed and text-box is cleared for the collection of next caption. *Numbered elements in caption interface*: 1. The prompt for the participants. 2. The text-box for the participants to input caption. 3. Image to be captioned. 4. Highlighted object in the scene.

A.3 Additional samples from generative semi-supervised model

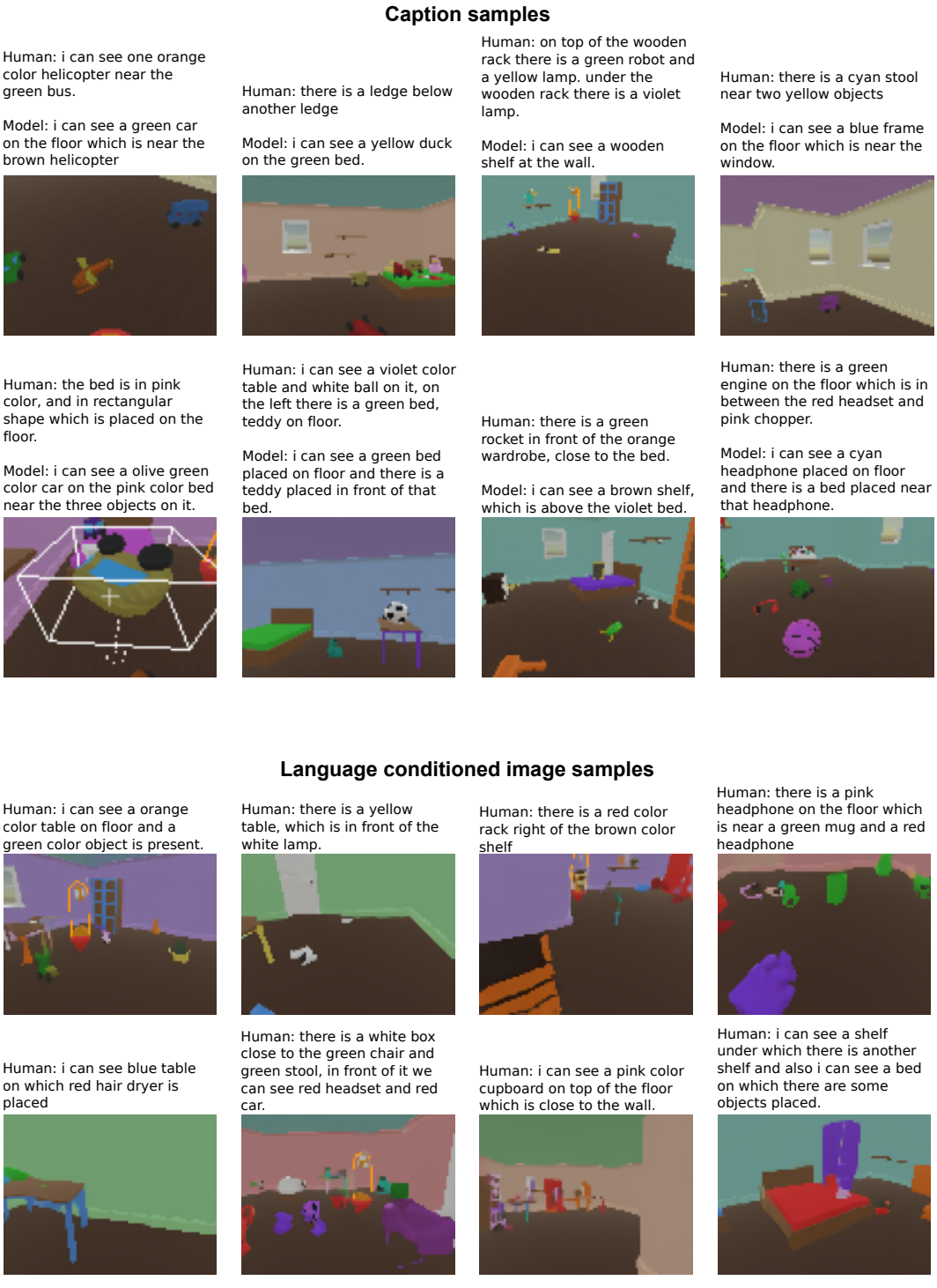

Figure A2: Additional caption and image samples from generative semi-supervised model.

A.4 Details about lift / ask color tasks

We modified the Playhouse environment [22] to create the lift drum and ask color drum tasks. These tasks were used for evaluation only, and the agent was never trained in these tasks. For these tasks, We first initialize a randomized playhouse environment as described in [22], which represents a randomized multi-room environment. We then spawn a drum object in the room where the agent avatar is spawned. The color of the drum object is randomly selected from the following list of 10 colors: "red", "yellow", "blue", "white", "green", "pink", "purple", "orange", "aquamarine", "magenta".

Lift task. In the lift task, the instruction "`Lift the drum.`" appears after a random delay of up to 10 seconds. A reward of 1 is given if the agent lifts any drum object, and the episode terminates after reward is emitted. If the agent lifts any other object, or times out after 2 minutes, the episode terminates with a reward of 0.

Ask about color task. In the ask about color task, the instruction "`What is the color of the drum?`" appears after a random delay of up to 10 seconds. If the agent emits language that matches the color of the drum, a reward of 1 is given, and the episode terminates. Otherwise if the agent outputs any other text, or times out after 2 minutes with no language output, the episode terminates with a reward of 0.

For each agent, we averaged rewards collected from $1,000$ episodes for each task. We also collected human scores on these tasks, and used it to normalize the agent reward to a range of $[0, 1]$. The human normalized score is reported in the manuscript.

A.5 Results on control objects.

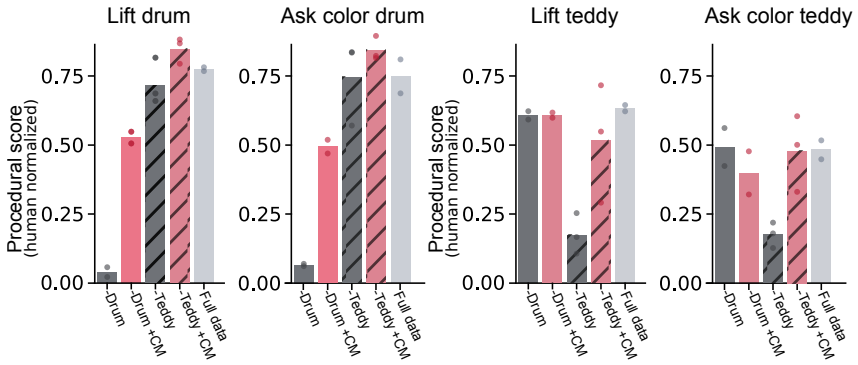

Figure A3: Performance on manipulating a control object. To confirm that our approach can work on any object, we performed additional experiments focusing on zero-shot generalization of the teddy bear object (agents labelled with "-Teddy"). Similarly to the drum experiments, the teddy bear object is excluded in the background interaction data and the agent has been trained without data manipulating teddy bear. We evaluated the agents in tasks that are similar to the lift and ask about color tasks described in Appendix A.4. In the teddy tasks, the drum object is replaced with the teddy bear object, and all instance of the word "drum" in the instructions are replaced with "teddy bear". We show that, similarly to the drum experiments, our caption matching loss and caption loss allow the agent to zero-shot manipulate the novel teddy bear objects (-Teddy +CM), compared to the baseline (-Teddy). Moreover, testing our agent on a known object (-Teddy +CM on drum, and -Drum +CM on teddy) shows that our method does not negatively affect the agent's ability to manipulate a known object.