

---

# Know your audience: specializing grounded language models with the game of Dixit

---

**Aaditya K Singh**

Gatsby Computational Neuroscience Unit  
University College London  
London, W1T 4JG  
aaditya.singh.21@ucl.ac.uk

**David Ding**

DeepMind  
London, UK  
fding@google.com

**Andrew Saxe**

Gatsby Computational Neuroscience Unit  
University College London  
London, W1T 4JG  
a.saxe@ucl.ac.uk

**Felix Hill**

DeepMind  
London, UK  
felixhill@google.com

**Andrew Kyle Lampinen**

DeepMind  
London, UK  
lampinen@deepmind.com

## Abstract

Effective communication requires adapting to the idiosyncratic common ground shared with each communicative partner. We study a particularly challenging instantiation of this problem: the popular game Dixit. We formulate a round of Dixit as a multi-agent image reference game where a (trained) speaker model is rewarded for describing a target image such that one (pretrained) listener model can correctly identify it from a pool of distractors, but another listener cannot. To adapt to this setting, the speaker must exploit differences in the common ground it shares with the different listeners. We show that finetuning an attention-based adapter between a CLIP vision encoder and a large language model in this contrastive, multi-agent setting gives rise to context-dependent *natural language* specialization from rewards only, without direct supervision. In a series of controlled experiments, we show that the speaker can adapt according to the idiosyncratic strengths and weaknesses of various pairs of different listeners. Furthermore, we show zero-shot transfer of the speaker’s specialization to unseen real-world data. Our experiments offer a step towards adaptive communication in complex multi-partner settings and highlight the interesting research challenges posed by games like Dixit. We hope that our work will inspire creative new approaches to adapting pretrained models.

## 1 Introduction

Human language use is communicative, and thus involves substantial adaptation to each conversational partner and context [1–4]. We can adapt our speech to complex social settings, with multiple partners and competing constraints, such as politeness vs. explicitness [5]. While current language and captioning models can increasingly imitate human language and scene descriptions [6–10], they generally do not explicitly adapt to a particular partner or attempt to satisfy multiple constraints.

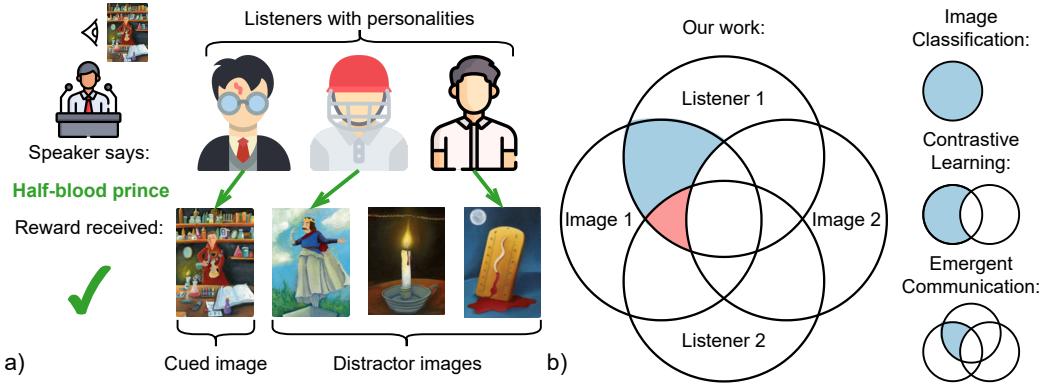


Figure 1: a) An example round of Dixit. The speaker uses a specialized caption (green), so that only one of the listeners (the one who has read Harry Potter) correctly identifies the image. This outcome is rewarded per the *some but not all* rule. b) How our work relates to past work. Image classification: models say something about an image. Contrastive learning: models say something about an image to distinguish it from others. Emergent communication: models say something about an image to distinguish it from others in a way one listener will understand. Our work: models say something about an image to distinguish it from others in a way one *but not another* listener will understand.

Here, we investigate training a model to satisfy a particularly challenging instantiation of constrained, adaptive, and grounded communication: the game of Dixit.

Dixit [11] is a popular multi-player game that requires adapting language to one’s audience. In each round, the “speaker” must describe a chosen image in language. Then, the rest of the players (two or more “listeners”) attempt to identify the target image from a pool of distractors. Importantly, the speaker is rewarded when *some but not all* listeners are able to correctly identify the image. This leads to the use of creative captions that target the common ground between a speaker and some (but not all) of the listeners. An example is illustrated in Figure 1a: the use of a Harry Potter reference to Professor Snape, the “half-blood prince”, as opposed to a more literal caption, such as “potions maker”, leads to the speaker receiving reward. For more details on the full Dixit game, we refer readers to prior work [12] introducing Dixit as a grand challenge for AI.

The *some-but-not-all* reward structure of Dixit encourages language specialization (from humans) in a grounded setting [12]. We take inspiration from this reward structure to formulate a setting where grounded language models can specialize their language without direct supervision, from rewards only. We build on prior work in emergent communication [13], and adapt a pretrained captioning model (the “speaker”) by finetuning a small fraction of its parameters to maximize a Dixit-inspired reward—this minimal adaptation helps reduce language drift [14]. We go beyond prior work by investigating a complex, not fully cooperative, setting (Figure 1b). The speaker’s goal is to communicate information to some listeners, but not others, by exploiting differences in listeners’ idiosyncratic “personalities”. We instantiate this setting by having a speaker model communicate with multiple (frozen) contrastive listener models. We design careful experiments, with datasets, metrics, and controls to ensure quantifiable assessment of language specialization.

We show that training a speaker in this Dixit-inspired setting can lead to *diverse* specialization of natural language across many pairs of listeners and datasets, with minimal language drift, and without direct supervision. From rewards, the speaker identifies and learns to cue to the difference between two listeners, which we call “listener subtraction”. For example, when trained with one listener which sees color, and another which sees only grayscale, the speaker learns to exclusively use color words—it exploits the specialized common ground it shares with the first listener. Furthermore, the speaker exhibits some zero-shot transfer of its language specialization from artificial to realistic datasets.

To summarize, our main contributions are: 1. We formulate Dixit as a multi-agent image reference game (Section 3.1). 2. We show that finetuning a pretrained grounded language model in this setting leads to the robust emergence of specialized common ground via listener subtraction, without substantial language drift (Section 4.1, 4.2). 3. This language specialization can transfer zero-shot from artificial to realistic datasets (Section 4.3). 4. We identify the key aspects of our approach through

a series of ablation experiments (Section 4.4). To our knowledge, our work is the first to consider natural language communication in a grounded, multi-listener setting, without direct supervision.

## 2 Related work

**Dixit.** Kunda and Rabkina [12] propose Dixit as a grand challenge for AI. Their paper discusses the full game of Dixit and the various interesting subproblems that would need to be solved (but does not attempt to solve them). Here, we focus on addressing the “Find a Phrase” subproblem—how the speaker should choose a phrase such that some but not all listeners correctly identify the image.

**Adapting (grounded) language models.** A large body of prior work focuses on adapting large pretrained language models [6, 7, 15, 16] to various tasks. Our work is similar in its focus on natural language specialization, but differs in that we do not use any supervised data. In that sense, our work is more similar to approaches to tuning pretrained models with reinforcement learning [17, 18]. Unlike those works, we focus on grounded communication. With respect to grounding, we take inspiration from the approach of prefix tuning [19] and use an image-to-prefix encoder to condition a pretrained language model on images, as in Tsimpoukelli et al. [7]. These models can adapt their language via prompting [6]; this approach is common in the low data regime, so we include it as a baseline.

**Emergent communication.** Finally, there is a relevant body of prior work on emergent communication in image reference games. We focus on works that consider natural language generation, which are most relevant to our setting. Cogswell et al. [13] focus on a speaker agent (Q-bot) that asks questions in dialogue with a single, pretrained-and-frozen listener (A-bot), to identify an image from a pool of distractors. Lazaridou et al. [20] consider speaker and listener agents in a more standard image reference game where a speaker must describe an image, and the single listener must choose the image from a pool of distractors. Both papers find the pretrain-then-finetune approach essential to maintaining natural language coherence, and we adopt a similar methodology. Like Cogswell et al. [13], we pretrain and freeze our listeners as our focus is on the speaker (and fixed listeners help maintain language grounding and avoid drift [cf. 14]). Unlike prior work, our setting involves simultaneous, natural language communication to multiple listeners and is not fully cooperative.

## 3 Methods

### 3.1 Multi-agent image reference game

We focus on a multi-agent image reference game inspired by a single round of Dixit. We frame this problem as a challenge of tuning a pretrained multimodal model to specialize its language, without direct supervision on how to specialize. Our setup (Figure 2) involves one speaker model communicating with multiple listener models about a target image. The speaker model receives the target image and outputs a caption. Each listener model receives the caption and the target image along with some (random) distractor images. Each listener model’s goal is to correctly identify the target image from the distractors, based on the speaker’s caption.

Following the some-but-not-all reward structure of the Dixit game, we reward the speaker model for *the difference in the (binary) accuracies of listeners 1 and 2 on a given set of images*. To avoid co-adaptation and pragmatic drift [cf. 20], we pretrain and freeze our listener models. Thus, only the speaker has learnable parameters.

### 3.2 Speaker model

For the speaker model (Figure 2a), we take inspiration from two past works on image captioning: the *Frozen* captioning model [7] and ClipCap [8]. We build the speaker architecture using several pretrained components, and adapt only a small part of the architecture to our setting.

Specifically, the speaker model is composed of a pretrained-and-frozen CLIP visual encoder [21], learned attentive (QKV) adapter layer, and a pretrained-and-frozen Transformer (decoder) language model [22, 23]. An input image is first passed through the visual encoder. The unpooled output of the encoder is flattened and passed into the adapter, a Perceiver-inspired cross attention layer [24]. The adapter outputs  $n$  tokens, which are fed as a prefix to the language model, which generates up to 32 tokens conditioned on this prefix. More architectural details can be found in Appendix A.

*The only parameters of this model that are ever trained are in the adapter.* We keep the pretrained CLIP encoder and language model frozen (to help prevent language drift), and pretrain the adapter parameters on the Conceptual Captions dataset [25] to give our speaker basic captioning abilities (see Appendix B.1). All our experiments start from this captioning-pretrained base speaker model. We finetune the adapter parameters for each experimental condition (we refer to this process as “training the speaker”). We do not have supervision for what to say when playing the game—much as a human player needs to adapt without being told precisely what to say. Instead, we use REINFORCE [26] to train the speaker’s adapter parameters, based on the rewards received for differences in the listener accuracies (see Section 3.1). Our choice to finetune only the adapter is in-line with prior works in emergent communication that use a frozen pretrained language model or visual encoder [13, 20].

### 3.3 Listener models

For our listener models (Figure 2b,c), we use the vision and language encoders of a pretrained ALIGN NFNet F5 model [27, 28]. For a set of input images and caption, we compute image-caption match scores for each image, and the listener selects the image with the highest alignment score.

To investigate the speaker’s ability to adapt to distinct listeners, we experimentally manipulate the listeners’ knowledge. Specifically, we use the same pretrained model for each listener, but manipulate each listener’s knowledge by applying distinct image transformations to the inputs. For listener 1, we do not transform the input images. For listener 2, we consider the following transformations:

1. Crop: Input images are cropped to the top-right  $112 \times 112$ , then resized back to  $224 \times 224$ .
2. Blur: Input images are Gaussian blurred with a radius of 25 pixels.
3. Grayscale: Input images are transformed to grayscale.

While listener “personalities” could differ along any linguistic, conceptual, and perceptual axes, we focus on these perceptual transformations because they enable quantitative assessments of language specialization. When listener 2 sees cropped images, we expect the speaker to cue objects outside the cropped region. When listener 2 sees blurred images, we expect the speaker to cue objects without the use of color, as distinct objects will not be visible to the second listener but color will still largely be visible. Conversely, when listener 2 sees grayscale, we expect the speaker to only use color words and to specifically stop referring to objects (which would provide signal to both listeners). We design controlled datasets for each of our experiments in which each of the relevant metrics can be measured.

### 3.4 Datasets and metrics

The main datasets we use are transformed versions of Fashion-MNIST [29]. Fashion-MNIST (FMNIST) is a dataset consisting of  $28 \times 28 \times 1$  grayscale images of different types of clothing items. We create a colored version (which we will refer to as cFMNIST), where each image is randomly colored with one of 8 colors. To test training with the crop listener, we then create a tiled version of cFMNIST (which we will refer to as tcFMNIST) where each image consists of two random cFMNIST

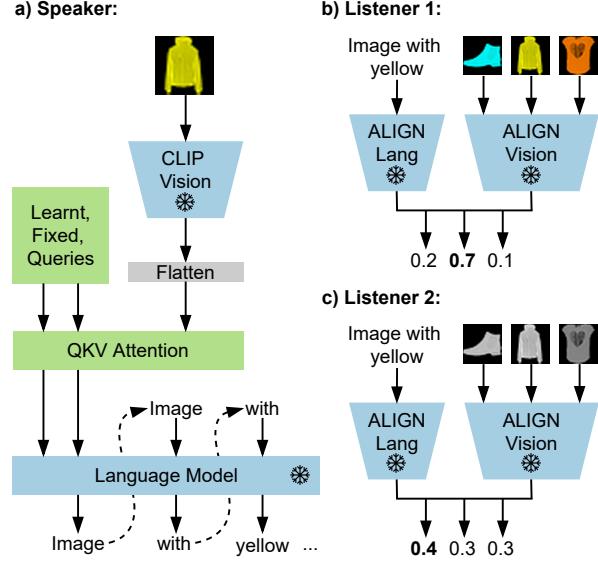


Figure 2: Our setup for cFMNIST and the grayscale transform. a) The speaker receives an image and generates a caption. b) Listener 1 receives unperturbed images and picks the best match to the caption (correctly). c) Listener 2 receives images with the grayscale transform applied and picks the best match to the caption (incorrectly). Then, the speaker receives a reward of 1 (the difference in the accuracies) which it uses to update its adapter parameters (green) via REINFORCE. All components in blue are pretrained and frozen.

images, one in the top-right, and one in the bottom-left. For more realistic images, we use the COCO dataset [30]. COCO images are resized so that the smallest dimension is 224, then a center square crop is taken. Example images for all dataset and transform pairs are shown in Figure 3.

For analysis metrics, we first evaluate the proportion of captions generated by the speaker on test images that had color words in them (“color prevalence”) and the proportion of captions that contained a clothing-related word (“FMNIST keyword prevalence”). Each of these metrics identifies language specialization. To ensure that the produced language remains image-relevant, we compute additional metrics measuring how often an object word used corresponded to an actual object in the image. Since the language model may know many synonyms, we used a set of synonyms when computing the object metrics, e.g., we would count “shoe” as a reference to a “boot”. These metrics are referred to as “object prevalence” for cFMNIST, and “bottom-left object prevalence” and “top-right object prevalence” for tcFMNIST. For colors, it’s difficult to enumerate all synonyms as the model often uses pairs of colors, e.g., “blue and white” for “cyan”. Instead, we found that a “color diversity” metric—which measures the speaker’s variation in color use—offers an effective (if slightly noisy) proxy for determining color relevance.

More specifics on datasets and metrics can be found in Appendices C, D, respectively.

### 3.5 Training

A training “episode” consists of the speaker receiving an image (observation), generating a caption (action), and receiving the difference in listener accuracies as a reward. As noted above, only the adapter in the speaker has trainable parameters, which are updated via REINFORCE [26]. We use nucleus sampling [31] to generate captions. We add a small reward penalty to incentivize shorter captions (a weight  $\lambda$  times the number of words in the caption). We use a fixed batch of 128 images, with 3 random distractors for each image drawn from the same batch. See Appendix B.2 for details.

## 4 Results

### 4.1 Emergence of specialization

By training a speaker to maximize the difference in accuracy between two listeners, we see strong language specialization across many listener pairs (Figure 3a-c, 4a-c). When the second listener sees only the top-right crop of the image, the speaker learns to cue to objects that the second listener cannot see (Figure 4a, bottom). Specifically, the first listener’s accuracy is relatively constant through learning, but the second listener’s accuracy decreases to chance level, in correspondence with the decrease in top-right object prevalence in the speaker’s captions. This decrease is accompanied by an *increase* in bottom-left object prevalence, which explains why the first listener remains accurate.

When the second listener sees only the blurred version of the image, the speaker learns to reduce its use of color words, while still referring to the target object. The overall learning dynamics are similar to the crop case: the first listener’s accuracy remains roughly constant, while the second drops to chance level, again paralleling the decrease in color prevalence in the speaker’s captions.

By contrast, when the second listener sees only a grayscale image, we see a rapid decrease in object prevalence and increase in color prevalence. However, this initial learning causes both listener

	Speaker and Listener 1 see:	Listener 2 sees:	Speaker output	
			Start of training:	End of training:
a)			a set of shoes and a t-shirt	a t-shirt
b)			the yellow jacket with a black hood	a hooded jacket
c)			the yellow jacket with a black hood	yellow
d)			person, a cyclist, is pictured in his new bike	a red and white

Figure 3: Example test images and speaker captions at the start and end of training for various dataset and listener pairs. By row, the speaker trains on a different dataset (a: tcFMNIST, b,c: cFMNIST, d: COCO), and the second listener sees a different transformed image (a: top-right crop, b: blurred, c,d: grayscale). Diverse language specialization emerges by the end of training. See Appendix H for more samples.

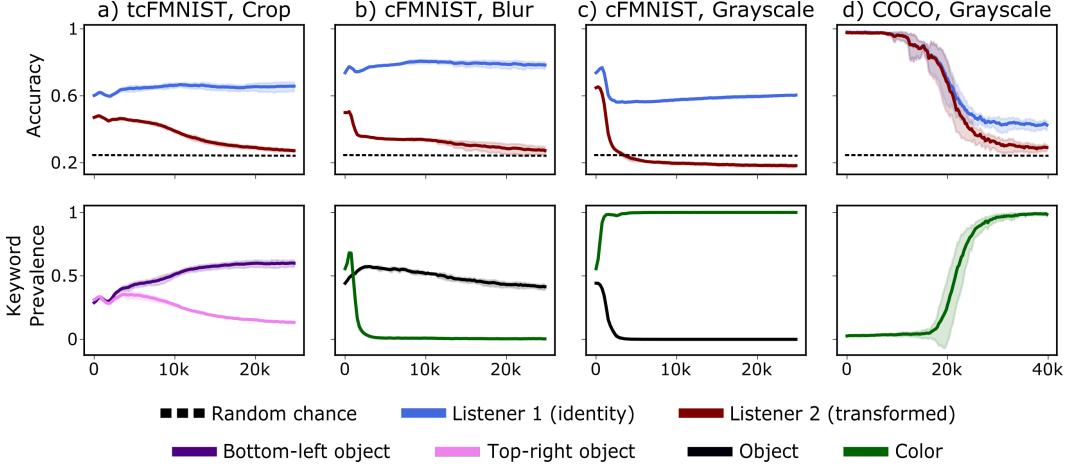


Figure 4: Accuracy and task-relevant metrics for many dataset and listener pairs. By column, the second listener sees a different transformed image (a: top-right crop, b: blurred, c,d: grayscale). The top row shows listener accuracies as the speaker learns to specialize. The bottom row shows prevalence of task-relevant keywords indicated by line color—note that object metrics measure both grounding and language specialization, while color metrics only measure specialization (see Section 3.4). Error bars show 95% confidence intervals (95% CIs) over 5 random seeds.

accuracies to drop, as the speaker greedily uses colors that may not be image relevant. Then, from iterations 2.5k to 25k, the speaker slowly learns to use more accurate colors for each image, and the first listener’s accuracy increases slowly but steadily (while the second listener stays at chance). At the end of training, color diversity is comparable with that produced by language prompts (see Section 4.4.3, Tables 1 and 15), indicating that the speaker is using diverse, image-relevant colors.

These experiments illustrate that the speaker model can optimize its reward in different ways depending on the pair of listeners—decreasing the second listener’s accuracy, increasing the first listener’s accuracy, or both. Each strategy produces meaningful, measurable changes in the speaker’s language.

Importantly, language drift is minimal—captions remain grounded by the above metrics, and show minimal structural drift (assessed by an independent language model’s likelihoods, Appendix E). For qualitative assessment, we show sample images, distractors, and speaker outputs in Appendix H.

## 4.2 Extending to real world data

We next apply our approach to a setting with real-world data—images from the COCO dataset [30]. The effect of most transforms is harder to quantify exactly, as the images contain many objects. Thus, we use the grayscale transform, which still offers usable metrics (color prevalence and diversity).

Figure 3d, 4d show model samples and the speaker’s training curves on COCO. As COCO images are far closer in distribution to Conceptual Captions (used to pretrain the speaker model), the initial accuracy of both listeners on the speaker’s captions is extremely high ( $\sim 98\%$ ). The speaker then “explores” different strategies to differentiate the listeners (iterations 0–20k). Just before 20k iterations, the speaker starts using colors slightly more often, then exhibits a near-stage-like transition to exclusively using colors. A qualitative investigation of individual sample captions on a set of images over training indicated that for each image, at some point during training, the speaker switches a standard caption (e.g., “person in front of a bicycle”) to a color-focused caption (e.g., “red and white”). Once the speaker only uses color for its captions, we see a significant difference between the individual listener accuracies. Notably both listener accuracies are far worse than at the start of training, as color alone is an imperfect distinguisher on COCO (and overspecifying colors may give away too much, since object category is correlated with color in real images). However, the speaker has successfully learned to specialize its language to exploit the difference between the two listeners. Furthermore, this language remains grounded, as evidenced by end-of-training color diversity being comparable to color diversity produced from language prompting (see Section 4.4.3, Tables 1 and 16).

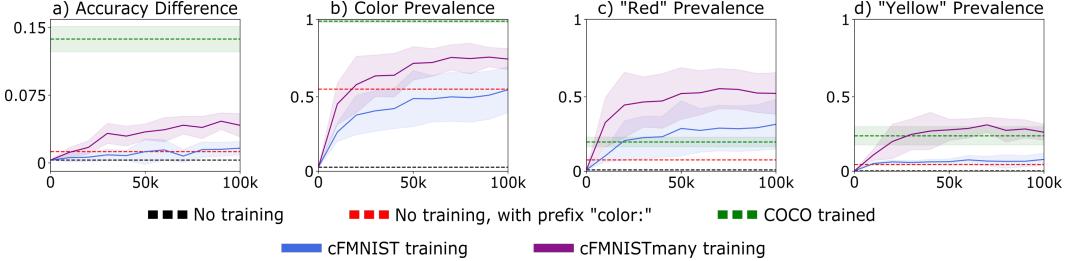


Figure 5: Zero-shot transfer results from cFMNIST to COCO and relevant baselines. Each subplot shows a different metric, with different lines corresponding to different conditions or baselines. The two conditions are training on cFMNIST (blue) or an augmented cFMNIST with 360 hues (purple). Various baselines are also included as dotted lines: base captioning model performance (black), base captioning model with explicit prompting (red), fully trained on COCO (green).

### 4.3 Zero-shot transfer

In this section, we explore zero-shot transfer of the language specialization from our simpler dataset (cFMNIST) to the more challenging setting of COCO. To do so, we trained a speaker in the grayscale transform setting on cFMNIST (Listener 1 sees the unperturbed image, Listener 2 sees the grayscale image) and tested its zero-shot behavior on COCO. As baselines, we consider the behavior of the pretrained captioning model (that the speaker is initialized to), and the behavior given the best natural-language prompt (see Section 4.4.3). We also compare to the behavior of an expert speaker trained to specialize on COCO (the end of training in Figure 4d).

We find significantly-above-baseline zero-shot transfer, with increasing use of color on COCO as the speaker is trained on FMNIST (Figure 5b). Furthermore, this specialization leads to meaningful differences in listener accuracies (Figure 5a). This specialization transfer from FMNIST to COCO performs better (across all four metrics in Figure 5) than baseline zero-shot approaches (the untuned captioning model), and on par with the best language prompt (see Section 4.4.3).

However, zero-shot behavior is far from perfect—the model loses some grounding relative to the experiments above. Zero-shot, the model strongly prefers “red” (Figure 5c), using it in nearly twice as many captions as the COCO-trained speaker, including even some non-red images (e.g., giraffes). To improve generalization, we therefore trained on a richer colored version of FMNIST, where image hues are sampled randomly from 360 equally-spaced options (as opposed to the 8-hue cFMNIST used above) – we refer to this dataset as cFMNISTmany. We find that this improves zero-shot transfer across all metrics, pushing it beyond the performance of the best explicit language prompt. Although this model has slightly increased “red preference”, its usage of other colors is also significantly better than the 8-hue-trained speaker. For example, in Figure 5d, we see that the cFMNISTmany-trained speaker uses yellow in a similar percentage of captions as a COCO-trained speaker. A better captioning model, that was adapted on a more diverse set of images, would likely transfer even better.

### 4.4 Ablations

In this section, we explore relevant ablations to our set-up to identify aspects that are crucial to our results. See Appendix F for full results (Tables 9-12).

#### 4.4.1 Single listener

In our original experiments, the speaker learns to specialize in the presence of multiple listeners, but are both listeners necessary? If the speaker just communicated with Listener 1, could it also specialize (perhaps differently across different random seeds)? To test this possibility, we train a speaker model to optimize the accuracy of just a single listener (so the reward per “episode” is binary 0/1 listener accuracy). Details of training are provided in Appendix B.3; results in Appendix F.

For the tcFMNIST case, we see that prevalence of keywords for both locations in the image increases fairly consistently, indicating a lack of specialization to image sub-regions. For cFMNIST, we find that training in the single listener regime leads to increased use of colors and correct keywords. Again, this behavior is fairly consistent, and we never see a differential specialization (to color or objects)

that full multi-listener training induces in our original experiments. For COCO, we see the largest difference between single-listener and multi-listener cases, as the single listener case doesn’t ever learn to utilize color as a consistent tool for distinguishing images.

#### 4.4.2 Non-contrastive reward

We next consider training with multiple listeners, but a non-contrastive reward. Namely, instead of optimizing for the difference in accuracy between listeners, we optimize directly for the difference in ALIGN match on the cued image. We call this approach “non-contrastive” as no distractor images are required. Details of training are provided in Appendix B.4; results in Appendix F.

We find that training in this setting does lead to some language specialization, but is worse than contrastive training in 3 out of 4 settings we consider. For the (unperturbed, crop) setting (Table 9), we see a reduced use of the top-right keyword, but only a modest increase in the use of the bottom-left keyword compared to the contrastive reward training. For the (unperturbed, blur) setting (Table 10), the non-contrastive reward actually outperforms the contrastive reward. The model specializes on using image-relevant keywords (70% of the time), and stops using colors in a useful way (colors are still used, but it’s always the same color as indicated by the diversity metric being 0, which provides no signal to the second listener). For the (unperturbed, grayscale) setting on both FMNIST and COCO, we observe strong specialization using the non-contrastive reward (as evidenced by low FMNIST keyword prevalence, and high color prevalence in Tables 11, 12). However, the captions in this case lose image relevance, as evidenced by significantly lower color diversity. Qualitatively, we observe the model using the same color for each image (which achieves more reward than baseline, but is obviously not desirable). Thus, overall, we find that the contrastive reward is beneficial to encouraging specialization while retaining image relevance in this multi-listener setting.

#### 4.4.3 Explicit prompting

Table 1: Relevant metrics for various caption prefixes. Full training provides superior language specialization compared to explicit prompting.

Caption Prefix	FMNIST metrics				COCO metrics		
	Accuracy Difference	Color Prevalence	Color Diversity	Object prevalence	Accuracy Difference	Color Prevalence	Color Diversity
No prefix	0.089	0.556	0.014	0.442	0.003	0.027	0.022
the color of this image is	0.174	0.901	<b>0.020</b>	0.000	-0.010	0.135	<b>0.036</b>
color:	0.118	0.971	0.004	0.111	0.013	0.541	0.030
colors in image:	0.048	0.999	0.001	0.000	0.019	0.335	0.018
Fully trained	<b>0.422</b>	<b>1.000</b>	0.015	<b>0.000</b>	<b>0.156</b>	<b>0.988</b>	0.033

Finally, we evaluate a simpler, commonly-used approach to model specialization: explicit prompting. These experiments also provide a baseline for our zero-shot results (in Section 4.3), as no training is involved. We created a variety of possible prefixes to elicit the desired specialization, as explicit prompting can be quite brittle. Full results are provided in Appendix G.

Table 1 shows the top three prompts and relevant metrics for the (unperturbed, grayscale) transform pair, on both FMNIST and COCO. Explicit prompting does lead to some specialization (when compared to the base captioning model, with no prefix), but fails to account for the robust specialization we observe in Section 4.1, 4.2. In fact, captioning prefixes can be quite brittle, as evidenced by the diverse behavior we observe from superficially similar prompts. For “colors in image:” and “color:”, the model has high color prevalence, but very low diversity. Qualitatively, samples of the model are of the form “color: green, black, white, blue, red ...” where the model lists all colors, with the first color typically corresponding to the true color of the image. For “color:” we also observe that the model doesn’t completely stop using nouns. For “the color of this image is” we observe slightly reduced color prevalence, but much stronger color diversity and very little usage of nouns.

On COCO, similar color prompts yield surprisingly large differences in color prevalence. We also observe “switching” behavior: per image, the speaker either ignores the caption prefix (e.g., “color: person and her son play in the water”) or specializes (e.g., “color: blue and white”). Qualitatively, this behavior mirrors the stage-like transition in the COCO-trained speaker, in Section 4.2.

## 5 Discussion

We have demonstrated a method for specializing a grounded language model without any *direct supervision*, by finetuning a small fraction of its parameters in a complex multi-agent setting. We have shown that this approach enables diverse types of specialization with minimal language drift (Section 4, Appendix E), and have identified which aspects of this approach are essential (Section 4.4). We believe that our work offers a novel perspective on adapting models to new settings without any supervised data. Below, we discuss some potential implications, accompanying ethical considerations and limitations, and highlight the future work needed to develop agents that can fully play Dixit.

Our approach to specializing a grounded language model rewards the speaker for using differences in the common ground it shares with two listener models. This formulation allows for flexible adaptation of models to complex tasks without supervision. For example, our approach could potentially be used to diagnose biases in one contrastive listener model by comparing to another (e.g., trained on a different dataset), an area of considerable interest [32]. A speaker optimized for the difference between two contrastive models could lead to an identification of biases present in one and not another, without the need for manually curated word lists. This approach could extend prior work on “red teaming” [33] whereby one model is used to identify biases in another. Our approach could also potentially be used to improve speakers without supervision. For example, we could potentially flip the above scenario and debias a speaker by rewarding it for communicating better to an unbiased than a biased listener. Alternatively, training against listeners of different skill could potentially improve language quality without relying on (costly) human annotations [e.g. 17].

Furthermore, our approach permits easy extension to other input modalities. Our pretrained-and-frozen visual encoder could be replaced with a modality-specific encoder (e.g., for audio [34]) or even just language input (which has shown surprising flexibility in expressing diverse tasks [6]), to extend to other domains. Beyond language production, the general competitive-cooperative paradigm we study has seen increased interest in areas like RL (e.g., in multi-agent games [35] and using the difference of predictors for skill learning [36]), so we hope that our insights could lead to useful perspectives on adapting to complex objectives in broader settings as well.

In the long term, we envision our work potentially contributing to personalized AI, for example an AI assistant adapted to each user. Humans are known to interact preferentially with others who share *rare preferences* [37]. Thus, an assistant which personalizes its language to individual users—e.g., using Harry Potter references when talking to a Harry Potter fan—would likely be preferred to one that uses the standardized language found in typical supervised datasets. Personalization could also be useful for other kinds of communication, such as explanations—e.g., adapting feedback to be helpful for a particular student based on their idiosyncratic knowledge [cf. 38]. Critically, our approach enables such adaptation through interaction, without explicit language supervision.

### 5.1 Ethical considerations

As discussed above, applications for this work would train speakers using differences between listener models with different weights or even architectures. A potential risk of adapting speakers via this approach is that the speaker might pick up biases that one listener has that are not part of the intended specialization—an undesirable quality. For example, biases might include representational biases resulting from stereotyping [39]. One possible approach to mitigating this would be to explicitly include biased listeners, but always penalize the speaker for the biased listeners’ accuracy. Another approach could be debiasing the underlying (frozen) language model [40, 41], as the speaker’s propensity for producing offensive language is inherited from this underlying model. Undesirable biases may also result from the representational prevalence in the training data. Cultural references that are well-represented in the training data may see more adaptation than references belonging to groups which are underrepresented. To address this, future work could seek to finetune models to test performance across groups. Long-term, we believe that strategies like those detailed by Weidinger et al. [42] will be necessary to mitigate risks of large language models (and models such as ours that utilize them).

## 5.2 Limitations

In the present work, we only considered settings with two listeners (for ease of quantitative assessment), so additional challenges may be present when extending to three or more listeners. In addition, the current work assumes that listeners exist that differ along the axis of desired specialization; finding such listeners might be challenging. Extending prior work with procedurally-generated listener populations [43] from using attribute vectors to natural language could offer a route to such diverse listeners. Furthermore, despite the overall preservation of language (Appendix E), we qualitatively observed some degradation in some runs (e.g., an added “and” at the end of captions). Introducing a weighted KL-divergence loss term to the pretrained captioning model likelihood (as done by Lazaridou et al. [20]) might further improve specialized language quality.

## 5.3 Towards Dixit

Finally, we highlight the inspiration we took from the *Dixit* game, and the future work necessary to fully play it. Our use of a contrastive, Dixit-like reward (based on some-but-not-all listeners identifying a target) is crucial to achieving language specialization while retaining image relevance (Section 4.4.2). Yet there are several key aspects of the full game of Dixit that we have not yet addressed—we emphasize that Dixit is a grand challenge for AI, as Kunda and Rabkina [12] suggest.

For example, human players adapt their language from a small number of interactions, while our speaker trains for thousands of iterations. One potential path to overcoming this limitation would be to combine our work with concurrent progress in visual language models. Alayrac et al. [9] introduce a model, Flamingo, which can rapidly adapt to different visual language tasks in a few-shot setting. Notably, Flamingo uses similar architectural components to this work (e.g., a Perceiver-inspired adapter layer). Using such a high-quality base model as the speaker might enable reward-driven adaptation from just a few interactions, an essential element of Dixit gameplay.

To conclude, our work takes a step towards agents that play Dixit, and opens exciting future directions. We hope that this work will inspire further research on settings like Dixit, and help to enhance the capabilities and adaptability of grounded language models and communicative agents more broadly.

## Acknowledgements

The authors would like to thank Frederic Besse, Denis Teplyashin, and Maria Tsimpoukelli for advice on the engineering aspects of the work. We would also like to thank Angeliki Lazaridou, DJ Strouse, Stephanie Chan, and Oriol Vinyals for useful discussions and feedback on the draft.

The authors also thank Flaticon.com for providing free access to various icons used in Figures 1 and 2. We also thank the creators of Dixit [11] for creating an inspiring game and the images used in Figure 1.

## References

- [1] Herbert H Clark and Deanna Wilkes-Gibbs. Referring as a collaborative process. *Cognition*, 22(1):1–39, 1986.
- [2] Herbert H Clark. *Using language*. Cambridge university press, 1996.
- [3] Robert XD Hawkins, Noah D Goodman, and Robert L Goldstone. The emergence of social norms and conventions. *Trends in cognitive sciences*, 23(2):158–169, 2019.
- [4] Robert D Hawkins, Michael Franke, Michael C Frank, Adele E Goldberg, Kenny Smith, Thomas L Griffiths, and Noah D Goodman. From partners to populations: A hierarchical bayesian account of coordination and convention. *Psychological Review*, 2022.
- [5] Erica J Yoon, Michael Henry Tessler, Noah D Goodman, and Michael C Frank. Talking with tact: Polite language as a balance between kindness and informativity. In *Proceedings of the 38th Annual Conference of the Cognitive Science Society*, pages 2771–2776. Cognitive Science Society, 2016.
- [6] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel

- Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020.
- [7] Maria Tsimpoukelli, Jacob Menick, Serkan Cabi, S. M. Ali Eslami, Oriol Vinyals, and Felix Hill. Multimodal few-shot learning with frozen language models, 2021.
  - [8] Ron Mokady, Amir Hertz, and Amit H. Bermano. Clipcap: CLIP prefix for image captioning. *CoRR*, abs/2111.09734, 2021. URL <https://arxiv.org/abs/2111.09734>.
  - [9] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katie Millican, Malcolm Reynolds, Roman Ring, Eliza Rutherford, Serkan Cabi, Tengda Han, Zhitao Gong, Sina Samangooei, Marianne Monteiro, Jacob Menick, Sebastian Borgeaud, Andrew Brock, Aida Nematzadeh, Sahand Sharifzadeh, Mikolaj Binkowski, Ricardo Barreira, Oriol Vinyals, Andrew Zisserman, and Karen Simonyan. Flamingo: a visual language model for few-shot learning, 2022. URL <https://arxiv.org/abs/2204.14198>.
  - [10] Ahmed Elhagry and Karima Kadaoui. A thorough review on recent deep learning methodologies for image captioning. *CoRR*, abs/2107.13114, 2021. URL <https://arxiv.org/abs/2107.13114>.
  - [11] JL Roubira. Dixit.[board game], 2008.
  - [12] Maithilee Kunda and Irina Rabkina. Creative captioning: An ai grand challenge based on the dixit board game, 2020.
  - [13] Michael Cogswell, Jiasen Lu, Rishabh Jain, Stefan Lee, Devi Parikh, and Dhruv Batra. Dialog without dialog data: Learning visual dialog agents from VQA data, 2020.
  - [14] Jason Lee, Kyunghyun Cho, and Douwe Kiela. Countering language drift via visual grounding. *arXiv preprint arXiv:1909.04499*, 2019.
  - [15] Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *CoRR*, abs/2106.09685, 2021. URL <https://arxiv.org/abs/2106.09685>.
  - [16] Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. Fine-tuning language models from human preferences, 2020.
  - [17] Nisan Stiennon, Long Ouyang, Jeffrey Wu, Daniel Ziegler, Ryan Lowe, Chelsea Voss, Alec Radford, Dario Amodei, and Paul F Christiano. Learning to summarize with human feedback. *Advances in Neural Information Processing Systems*, 33:3008–3021, 2020.
  - [18] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*, 2022.
  - [19] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. *CoRR*, abs/2101.00190, 2021. URL <https://arxiv.org/abs/2101.00190>.
  - [20] Angeliki Lazaridou, Anna Potapenko, and Olivier Tieleman. Multi-agent communication meets natural language: Synergies between functional and structural language learning. *CoRR*, abs/2005.07064, 2020. URL <https://arxiv.org/abs/2005.07064>.
  - [21] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020, 2021. URL <https://arxiv.org/abs/2103.00020>.
  - [22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *CoRR*, abs/1706.03762, 2017. URL <http://arxiv.org/abs/1706.03762>.

- [23] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. Training compute-optimal large language models. *CoRR*, abs/2203.15556, 2022. URL <https://arxiv.org/abs/2203.15556>.
- [24] Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Daniel Zoran, Andrew Brock, Evan Shelhamer, Olivier J. Hénaff, Matthew M. Botvinick, Andrew Zisserman, Oriol Vinyals, and João Carreira. Perceiver IO: A general architecture for structured inputs & outputs. *CoRR*, abs/2107.14795, 2021. URL <https://arxiv.org/abs/2107.14795>.
- [25] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of ACL*, 2018.
- [26] Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8(3–4):229–256, may 1992. ISSN 0885-6125. doi: 10.1007/BF00992696. URL <https://doi.org/10.1007/BF00992696>.
- [27] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. *CoRR*, abs/2102.05918, 2021. URL <https://arxiv.org/abs/2102.05918>.
- [28] Andrew Brock, Soham De, Samuel L. Smith, and Karen Simonyan. High-performance large-scale image recognition without normalization. *CoRR*, abs/2102.06171, 2021. URL <https://arxiv.org/abs/2102.06171>.
- [29] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, abs/1708.07747, 2017. URL <http://arxiv.org/abs/1708.07747>.
- [30] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014. URL <http://arxiv.org/abs/1405.0312>.
- [31] Ari Holtzman, Jan Buys, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. *CoRR*, abs/1904.09751, 2019. URL <http://arxiv.org/abs/1904.09751>.
- [32] Sandhini Agarwal, Gretchen Krueger, Jack Clark, Alec Radford, Jong Wook Kim, and Miles Brundage. Evaluating CLIP: towards characterization of broader capabilities and downstream implications. *CoRR*, abs/2108.02818, 2021. URL <https://arxiv.org/abs/2108.02818>.
- [33] Ethan Perez, Saffron Huang, H. Francis Song, Trevor Cai, Roman Ring, John Aslanides, Amelia Glaese, Nat McAleese, and Geoffrey Irving. Red teaming language models with language models. *CoRR*, abs/2202.03286, 2022. URL <https://arxiv.org/abs/2202.03286>.
- [34] Hendrik Purwins, Bo Li, Tuomas Virtanen, Jan Schlüter, Shuo-Yiin Chang, and Tara N. Sainath. Deep learning for audio signal processing. *CoRR*, abs/1905.00078, 2019. URL <http://arxiv.org/abs/1905.00078>.
- [35] Max Jaderberg, Wojciech M. Czarnecki, Iain Dunning, Luke Marris, Guy Lever, Antonio García Castañeda, Charles Beattie, Neil C. Rabinowitz, Ari S. Morcos, Avraham Ruderman, Nicolas Sonnerat, Tim Green, Louise Deason, Joel Z. Leibo, David Silver, Demis Hassabis, Koray Kavukcuoglu, and Thore Graepel. Human-level performance in 3d multiplayer games with population-based reinforcement learning. *Science*, 364(6443):859–865, 2019. doi: 10.1126/science.aau6249. URL <https://www.science.org/doi/abs/10.1126/science.aau6249>.

- [36] Kate Baumli, David Warde-Farley, Steven Hansen, and Volodymyr Mnih. Relative variational intrinsic control. *CoRR*, abs/2012.07827, 2020. URL <https://arxiv.org/abs/2012.07827>.
- [37] Natalia Vélez, Sophie Bridgers, and Hyowon Gweon. The rare preference effect: Statistical information influences social affiliation judgments. *Cognition*, 192:103994, 2019. ISSN 0010-0277. doi: <https://doi.org/10.1016/j.cognition.2019.06.006>. URL <https://www.sciencedirect.com/science/article/pii/S0010027719301672>.
- [38] Lisa Wang, Angela Sy, Larry Liu, and Chris Piech. Learning to represent student knowledge on programming exercises using deep learning. *International Educational Data Mining Society*, 2017.
- [39] Paul Pu Liang, Chiyu Wu, Louis-Philippe Morency, and Ruslan Salakhutdinov. Towards understanding and mitigating social biases in language models. In *International Conference on Machine Learning*, pages 6565–6576. PMLR, 2021.
- [40] Timo Schick, Sahana Udupa, and Hinrich Schütze. Self-Diagnosis and Self-Debiasing: A Proposal for Reducing Corpus-Based Bias in NLP. *Transactions of the Association for Computational Linguistics*, 9:1408–1424, 12 2021. ISSN 2307-387X. doi: 10.1162/tacl\_a\_00434. URL [https://doi.org/10.1162/tacl\\_a\\_00434](https://doi.org/10.1162/tacl_a_00434).
- [41] Nicholas Meade, Elinor Poole-Dayan, and Siva Reddy. An empirical survey of the effectiveness of debiasing techniques for pre-trained language models. *CoRR*, abs/2110.08527, 2021. URL <https://arxiv.org/abs/2110.08527>.
- [42] Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, Zac Kenton, Sasha Brown, Will Hawkins, Tom Stepleton, Courtney Biles, Abeba Birhane, Julia Haas, Laura Rimell, Lisa Anne Hendricks, William S. Isaac, Sean Legassick, Geoffrey Irving, and Iason Gabriel. Ethical and social risks of harm from language models. *CoRR*, abs/2112.04359, 2021. URL <https://arxiv.org/abs/2112.04359>.
- [43] Rodolfo Corona Rodriguez, Stephan Alaniz, and Zeynep Akata. Modeling conceptual understanding in image reference games. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL <https://proceedings.neurips.cc/paper/2019/file/df308fd90635b28d82558cf580c73ed9-Paper.pdf>.
- [44] Taku Kudo and John Richardson. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *CoRR*, abs/1808.06226, 2018. URL <https://arxiv.org/abs/1808.06226>.
- [45] James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL <http://github.com/google/jax>.
- [46] Tom Hennigan, Trevor Cai, Tamara Norman, and Igor Babuschkin. Haiku: Sonnet for JAX, 2020. URL <http://github.com/deepmind/dm-haiku>.
- [47] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL <http://arxiv.org/abs/1412.6980>.
- [48] Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. Zero: Memory optimizations toward training trillion parameter models. *CoRR*, abs/1910.02054, 2019. URL <https://arxiv.org/abs/1910.02054>.
- [49] Ronald J. Williams and David Zipser. A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, 1(2):270–280, 1989. doi: 10.1162/neco.1989.1.2.270.

- [50] R. Keys. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(6):1153–1160, 1981. doi: 10.1109/TASSP.1981.1163711.

## A Speaker Model Details

We provide further detail on our speaker model architecture.

The pretrained-and-frozen CLIP visual encoder takes in images of dimension  $224 \times 224 \times 3$ , which we denote  $x$ . We take the pre-spatial-pool output from the model, which has dimensions  $16 \times 16 \times 768$ , and flatten this (across spatial dimensions) to an output  $E(x)$ , with dimension  $256 \times 768$ . For the specific encoder model, we experimented with a few (pretrained) options (see Appendix B.1).

The attention-based (QKV) adapter takes in this visual embedding and transforms it into  $n$  token embeddings that can be used as a prefix to prompt the language model. The trainable parameters of this layer are  $Q$  (of dimension  $n \times d$ ),  $W_K$  (of dimension  $768 \times d$ ), and  $W_V$  (of dimension  $768 \times 2048$ ). The weights  $W_K, W_V$  are used to compute  $K = E(x)W_K, V = E(x)W_V$ . The fixed queries are then used to attend to these input-dependent keys and values. The output of the adapter layer is thus  $A(x) = \text{softmax}(QK^\top / \sqrt{d})V$  (of dimension  $n \times 2048$ ).

We feed in  $A(x)$  as  $n$  prefix embeddings of dimension 2048 to a pretrained-and-frozen causal Transformer to generate up to 32 tokens (with early termination if the EOS token is produced). We follow the recommended architecture parameters presented in Hoffmann et al. [23] for a 1.4B parameter model: our transformer has 24 layers, model dimensionality of 2048, and 16 heads. We use a SentencePiece tokenizer with a vocabulary size of 32000 [44].

## B Training details

In this section, we provide training details, hyperparameter search details, and the hyperparameters we used for our final results. All models were implemented in Python using JAX [45] and Haiku [46]. Training was distributed over 16 TPUs (v3), and all experiments used a batch size of 128 (unless specified otherwise). The optimizer for all experiments is Adam [47], with  $\beta_1 = 0.9, \beta_2 = 0.95$ . We use ZeRO stage-one parameter sharding [48].

### B.1 Speaker model pretraining on captioning data

We pre-trained our speaker model using supervised cross-entropy loss (with teacher forcing [49]) on the Conceptual Captions dataset [25] which consists of paired image-caption data.

Dataset images were augmented using random crops. All experiments were run with a batch size of 512 for 500000 steps. Training was distributed over 16 TPUs (v3) in a data parallel fashion.

Hyperparameters we searched over were:

- Base CLIP model: We tried the smaller CLIP ViT-B/32 and the larger CLIP ViT-L/14 model.
- Positional embeddings: We experimented with adding an absolute positional embedding to each of the  $16 \times 16$  unpooled outputs from the CLIP encoder.
- Learning rate: [1e-4, 3e-4, 1e-3, 3e-3, 1e-2, 3e-2, 1e-1, 3e-1]
- Dimension of QKV adapter ( $d$ ): [32, 64]
- Number of input tokens to frozen language model ( $n$ ): [8, 16, 32]

All hyperparameter selections were based on validation loss on the Conceptual Captions validation set. We found that the larger CLIP model, no additional positional embeddings, a learning rate of 3e-2,  $d = 32$ ,  $n = 32$ , worked best. In terms of the biggest factors, using the larger CLIP model made the biggest change, followed closely by learning rate.

We froze the best model, and that same model served as the starting point for all our experiments.

### B.2 Multi-listener, contrastive-reward training

We used the hyperparameters detailed in Table 2 for training our main models. These hyperparameters were found by grid searches over learning rate ([3e-5, 3e-4]), sampling temperature ([1,2]), nucleus size ([0.8, 1]), and caption length penalty ([1e-5, 3e-5, 1e-4, 3e-4, 1e-3, 3e-3, 1e-2, 3e-2]). We found

Table 2: Hyperparameters for the main setting (multi-listener, contrastive-reward).

Parameter	Crop	<b>FMNIST</b>		<b>COCO</b>
		Blur	Grayscale	COCO Grayscale
Learning rate	3e-4	3e-4	3e-4	3e-4
Sampling temperature	1	1	1	1
Nucleus Size	0.8	0.8	1	1
Caption length penalty ( $\lambda$ )	3e-3	3e-3	3e-3	3e-4

that caption length penalty was crucial to tune to make sure it didn’t dominate the reward at the start of training.

All of our final training results are run on 5 different random seeds, from which 95% confidence intervals are calculated and shown (in figures and tables).

### B.3 Single listener, contrastive-reward training

Table 3: Hyperparameters for the single listener, contrastive-reward baseline.

Parameter	cFMNIST	tcFMNIST	COCO
Learning rate	3e-4	3e-4	3e-4
Sampling temperature	1	1	1
Nucleus Size	0.8	0.8	1
Caption length penalty ( $\lambda$ )	1e-5	1e-5	1e-5

For this baseline, we train the speaker to just maximize the binary accuracy of the single unperturbed listener. We train in 3 settings: on cFMNIST, on tcFMNIST, and on COCO. To make a fair comparison, we performed a hyperparameter search for each case (final hyperparameters shown in Table 3). Specifically, we searched over learning rate ([3e-5, 3e-4]), nucleus size ([0.8, 1]), and caption length penalty ([1e-5, 1e-4, 1e-3, 1e-2]). We also found that early stopping and batch-based baseline subtraction [26] improved performance (in terms of reward on unseen images), so we used both of these techniques for all results reported. On each setting, we trained 5 random seeds for up to 25000 iterations on FMNIST and up to 40000 iterations on COCO, early stopped each one, and computed evaluation metrics on those checkpoints.

### B.4 Multi listener, non-contrastive-reward training

Table 4: Hyperparameters for the multi-listener, non-contrastive-reward baseline.

Parameter	Crop	<b>FMNIST</b>		<b>COCO</b>
		Blur	Grayscale	COCO Grayscale
Learning rate	3e-4	3e-4	3e-4	3e-4
Sampling temperature	1	1	1	1
Nucleus Size	0.8	0.8	0.8	0.8
Caption length penalty ( $\lambda$ )	3e-6	3e-6	3e-6	3e-4

For this baseline, we train the speaker to just maximize the difference in the ALIGN match scores that listeners assign to the cued image. In this setup, the listeners only need the caption and cued image, which is why we call it non-contrastive. We train this baseline in all four settings that we train our main (multi-listener, contrastive-reward) model in. We perform hyperparameter over nucleus size ([0.8, 1]) and caption length penalty ([1e-7, 1e-6, 3e-6, 1e-5, 3e-5, 1e-4, 3e-4, 1e-3]), with optimal hyperparameters reported in Table 4.

## C Dataset details

To construct the cFMNIST dataset, we convert FMNIST images to HSV. Since the images are originally grayscale, the saturation is always 0. We set the saturation to 1, and set the hue to one of

Table 5: Color and Hues for cFMNIST.

Color:	Red	Orange	Yellow	Green	Cyan	Blue	Purple	Pink
Hue:	$\frac{0}{360}$	$\frac{30}{360}$	$\frac{60}{360}$	$\frac{120}{360}$	$\frac{180}{360}$	$\frac{240}{360}$	$\frac{270}{360}$	$\frac{300}{360}$

the 8 colors shown in Table 5. Then, we convert back to RGB images and resize to spatial dimensions  $224 \times 224$  using cubic upsampling [50], then clip all values to the range [0,1].

For tcFMNIST, we randomly place a cFMNIST image in the bottom left and in the top-right of the image. This choice of tiling (as opposed to top-left, bottom-right, or some other combination) was chosen to avoid an inherent bias towards cuing for the top-left object that we observed in our base captioning model.

For each dataset, we construct a fixed test set to use for equal comparison across all conditions. For contrastive-reward experiments, we also fix the distractors during test time for consistency.

We normalize all images according to the normalization that was originally used for the off-the-shelf image encoders we use (for consistency). Specifically, for inputs to the speaker, which pass through CLIP’s vision encoder, the normalization mean and standard deviation values are (0.481, 0.458, 0.408) and (0.269, 0.261, 0.278), respectively. For inputs to the listeners, which pass through ALIGN’s vision encoder, the normalization mean and standard deviation values are (0.485, 0.456, 0.406) and (0.229, 0.224, 0.225), respectively.

### C.1 Licenses

We use FMNIST under the MIT license, COCO under the creative commons license, and Conceptual Captions under its ad-hoc license (for which we thank Google LLC). We make use of the GPT3 [6] beta (for quantification of structural drift, see Appendix E) under the Apache License.

## D Evaluation metric details

We have two types of metrics: color metrics and keyword metrics. Color metrics apply on both FMNIST-based datasets and COCO, while keyword metrics are only used on FMNIST-based dataset (where we know exactly where and what the objects are by construction). We want our metrics to be able to diagnose language specialization, and also check that the captions are still grounded (we’ll refer to this as “image relevance” of captions).

### D.1 Color metrics

All color metrics utilize the following set of 16 colors:

red, orange, yellow, green, blue, indigo, violet, purple, cyan, magenta,  
pink, brown, black, white, gray, grey

For measuring language specialization to colors, we define *color prevalence*: the fraction of captions at test time containing at least one word from the above list.

For measuring image relevance, we use a proxy metric as its often hard to define what it means for a color to be correct (e.g., if an image is “cyan” and the model says “blue and white”). The proxy metric we use is *color diversity*. To compute color diversity we calculated TF-IDF vectors for each caption using only the color terms above. Then, we calculated the trace of the covariance matrix (which has dimension  $16 \times 16$ ) of these vectors. This metric has the desirable property where if a color appears in nearly all captions, it will be scaled down by the IDF term and so will its contribution to the trace of the covariance matrix.

We found this to be a good proxy metric as loss of image relevance in color specialization cases corresponding to the model choosing just a few colors (often just one) and using them to describe all images. To not confound our main results, we qualitatively looked at how well this metric corresponded to loss of image relevance on the many language prompts detailed in Appendix G. We found that color diversity was a noisy, but useful, metric for determining image relevance of these

prompts. For example, a simpler metric like checking the number of unique captions the model uses fails since, given some language prompts, the speaker would produce long strings of colors, often in slightly different orders (e.g. “color: green, black, white, blue, red ...” vs. “color: red, black, white, blue, green ...”). Furthermore, we note that color diversity is not a perfect metric, so subtle differences (on the order of  $\pm 0.005$ ) should not be considered significant. We mainly use color diversity to classify runs where image relevance was completely lost (color diversity going to 0), or image relevance was retained (color diversity staying near that of the best language prompts and above that at the start of training).

## D.2 Keyword metrics

Table 6: FMNIST keyword sets. We refer to the set of all words in this table as FMNIST related keywords.

Label	Category name	Added synonyms
0	t-shirt	top, t-shirts, shirt, shirts
1	trouser	trousers, pants
2	pullover	sweater, hoodie, sweaters, hoodies
3	dress	dresses
4	coat	coats, jacket, jackets
5	sandal	high heels, heels, shoe, shoes
6	shirt	shirts
7	sneaker	sneakers, shoe, shoes, running shoe
8	bag	purse, backpack, bags, purses
9	ankle boot	boot, shoe, shoes, boots

For measuring language specialization to objects, we define *FMNIST keyword prevalence*: the fraction of captions at test time containing at least one FMNIST related keyword.

To measure image relevance, we utilize the ground-truth labels from FMNIST, supplemented with synonyms. Specifically, we measure *object prevalence*: the fraction of captions at test time that contain at least one keyword corresponding to the ground truth label of the object in, the image according to Table 6. We found that allowing synonyms was essential, as the captioning model heavily prefers some clothing words over others (e.g., sandals are almost always just called shoes by the model). For tcFMNIST, we similarly define *bottom-left object prevalence* and *top-right object prevalence* to measure what fraction of captions refer to each region of the image.

## E Language drift quantification

Prior work in emergent communication [20] establishes three types of language drift that may occur when adapting language from rewards, without direct supervision: structural drift (how “language-like” are captions), semantic drift (how “image-relevant” are captions), and pragmatic drift (how “human-interpretable” are captions). While the focus of our work is on language specialization, it is important to investigate to what extent our approach is resulting in language drift.

For semantic and pragmatic drift, we note that our main metrics (see Section 3.4 and Appendix D) measure human interpretability as well as image relevance. For example, our object metrics measure whether the model is referring to objects using the category name or valid synonyms, and our color diversity metric is able to differentiate settings with significant semantic drift (e.g., the non-contrastive baseline, see Section 4.4.2) from settings without significant semantic drift (e.g., our main results, see Figures 3 and 4). Our method preserves these metrics, presumably since the grounded task with frozen listeners, as well as the frozen components of the speaker, provide strong constraints on what language the speaker can use.

However, our main metrics do not adequately address the issue of structural drift. To measure how “language-like” our captions are, we therefore follow the approach of Lazaridou et al. [20] and evaluate the log-likelihood assigned to generated captions by an independent, pretrained language model (LM). Specifically, we use OpenAI’s Ada model, made available online through the GPT3 beta [6]. For comparison, we also show LM log-likelihoods for the ground truth human captions (for

cFMNIST, we procedurally generate these—e.g., “red pullover”), the LM likelihoods for the “best prompt” (see Appendix G), and the LM likelihoods for captions from the speaker before specialization training (the base speaker pretrained on captioning only). For reference, the “best” language prompts were chosen to maximize task-relevant metrics (see third column in Tables 13-16) except for the (cFMNIST, Grayscale) case where we use the third best (as the top two prefixes have very low color diversity—see discussion in Section 4.4.3). Results are shown in Table 7.

Table 7: Average language model *full-caption* log-likelihoods for ground truth, start-of-training, and end-of-training captions on test images.

	tcFMNIST, Crop	cFMNIST, Blur	cFMNIST, Grayscale	COCO, Grayscale
Ground truth	-32.77	<b>-15.20</b>	-15.20	-46.26
“Best” language prompt	-36.61	-44.61	-24.59	-29.49
Start of training	-33.33	-28.64	-28.64	<b>-28.44</b>
End of training	<b>-17.41</b>	-19.01	<b>-9.26</b>	-29.49

Table 8: Average language model *per-token* log-likelihoods for ground truth, start-of-training, and end-of-training captions on test images.

	tcFMNIST, Crop	cFMNIST, Blur	cFMNIST, Grayscale	COCO, Grayscale
Ground truth	-5.823	-7.666	-7.666	-4.772
“Best” language prompt	-4.680	-4.371	<b>-4.076</b>	-4.051
Start of training	<b>-4.160</b>	<b>-4.127</b>	-4.127	-4.463
End of training	-7.143	-8.383	-4.234	<b>-2.509</b>

Surprisingly, we see that in most of our runs, average caption likelihoods seem to increase as the speaker specializes. It appears that this effect may be driven by the length penalty—over training, the speaker produces shorter captions, which have higher likelihoods since they have fewer tokens. Qualitatively, we find that language drift can be fairly variable across different random seeds. For example, the COCO average is worse after training largely due to a single run, in which long repeated captions emerged (which have lower likelihoods than their un-repeated counterparts).

To evaluate this further, we computed the per-token likelihoods (Table 8). While these do show some decrease in likelihood in some cases, the captions at the end of training are generally of comparable likelihood to the ground truth. In this instance, we notice that the one COCO seed in which the captions repeated actually exhibits *greater* per-token likelihood—after a few repeats, the LM starts to estimate further repeats to be very likely. This could potentially be a concern for using LM likelihoods as a metric for structural drift more broadly [cf. 14]. However, in most runs our length penalty prevents repetitions or long captions, as noted above.

In summary, these results indicate that language has not drifted far in most conditions and for most random seeds. We attribute this to some of the same factors that help prevent other types of language drift: finetuning a small part of our speaker (just the adapter), keeping the listener models frozen (thus avoiding co-adaptation), using a length penalty (see above discussion), and using contrastive reward (crucial for combating semantic drift, see Section 4.4.2). If necessary, language drift could potentially be reduced further by using a KL-divergence loss to the distribution of outputs from the base captioning model (as done by Lazaridou et al. [20]).

## F Full ablation results

In this section we show the full quantitative ablation results, in Tables 9, 10, 11, and 12.

Table 9: Metrics (with 95% CI) for ablations on the tcFMNIST dataset and the (unperturbed, crop) pair of listener transformations. Both non-contrastive and full training lead to decreased use of top-right keyword, but only full training accurately specializes to using the bottom-left keyword.

Condition	Bottom-left object prevalence	Top-right object prevalence
No training	0.290	0.312
Single listener	$0.475 \pm 0.024$	$0.473 \pm 0.020$
Non-contrastive	$0.360 \pm 0.005$	<b><math>0.120 \pm 0.011</math></b>
Full training	<b><math>0.602 \pm 0.066</math></b>	$0.135 \pm 0.017$

Table 10: Relevant metrics (with 95% CI) for various ablations on the cFMNIST dataset and the (unperturbed, blur) pair of listener transformations. Non-contrastive rewards perform better than contrastive rewards in this setting, as evidenced by increased use of correct keywords, and decreased use of meaningful colors (diversity of colors goes to 0).

Condition	FMNIST keyword prevalence	Object prevalence	Color prevalence	Color diversity
No training	0.644	0.442	0.556	0.014
Single listener	$0.754 \pm 0.020$	$0.564 \pm 0.013$	$0.953 \pm 0.022$	$0.016 \pm 0.000$
Non-contrastive	<b><math>0.997 \pm 0.004</math></b>	<b><math>0.710 \pm 0.086</math></b>	$0.606 \pm 0.833$	<b><math>0.000 \pm 0.000</math></b>
Full training	$0.600 \pm 0.044$	$0.418 \pm 0.038$	<b><math>0.004 \pm 0.008</math></b>	$0.001 \pm 0.002$

Table 11: Relevant metrics (with 95% CI) for various ablations on the cFMNIST dataset and the (unperturbed, grayscale) pair of listener transformations. Full training performs best as color prevalence and diversity increase, while keyword prevalence decreases to 0.

Condition	Color prevalence	Color diversity	FMNIST keyword prevalence	Object prevalence
No training	0.556	0.014	0.644	0.442
Single listener	$0.953 \pm 0.022$	<b><math>0.016 \pm 0.000</math></b>	$0.754 \pm 0.020$	$0.564 \pm 0.013$
Non-contrastive	$1.000 \pm 0.000$	$0.005 \pm 0.003$	$0.000 \pm 0.000$	$0.000 \pm 0.000$
Full training	<b><math>1.000 \pm 0.000</math></b>	$0.015 \pm 0.002$	<b><math>0.000 \pm 0.000</math></b>	<b><math>0.000 \pm 0.000</math></b>

Table 12: Relevant metrics (with 95% CI) for various ablations on the COCO dataset and the (unperturbed, grayscale) pair of listener transformations. Full training performs best as it has highest color prevalence and diversity.

	Color prevalence	Color diversity
No training	0.027	0.022
Single listener	$0.102 \pm 0.080$	$0.030 \pm 0.007$
Non-contrastive	$0.932 \pm 0.256$	$0.018 \pm 0.005$
Full training	<b><math>0.988 \pm 0.023</math></b>	<b><math>0.033 \pm 0.027</math></b>

## G Extended results on caption prefixes

In Tables 13-16, we show all caption prefixes we experimented with, as well as the relevant metrics for each. For each table, we sort prefixes from worst (top of table) to best (bottom of table) based on the most task-relevant keyword metric (third column in each table). We added a column for “Score difference” which is the average difference in ALIGN match scores from each listener for the cued image (it corresponds to the reward that is seen in the non-contrastive reward baseline). Of course, no training occurs, as the caption prefixes just explore how well the speaker can do by just explicit prompting. We also use  $\_\!$  to indicate a space at the end of the caption. For most prefixes, we see a large difference between adding this space and not adding it, which is just another testament to how brittle prompt engineering can be.

Table 13: Various metrics for prompted generation of base captioning model on tcFMNIST. Accuracy and score difference are calculated on the (unperturbed, crop) pair of listeners.

Caption prefix	Accuracy difference	Score difference	Bottom-left object prevalence	Top-right object prevalence	FMNIST keyword prevalence
the bottom left of this picture is $\_\!$	0.0121	0.0061	0.0181	0.0193	0.0502
the bottom left of this image is $\_\!$	0.0052	0.0065	0.0241	0.0277	0.0794
in the bottom left of this image, there is $\_\!$	0.0512	0.0112	0.0663	0.0719	0.1688
bottom left of this image: $\_\!$	0.0155	0.0096	0.0711	0.0798	0.2604
the bottom left of this image shows $\_\!$	0.0338	0.0137	0.0993	0.1212	0.3179
bottom left: $\_\!$	0.0418	0.0439	0.1154	0.1274	0.4156
in the bottom left of this image, there is	0.0681	0.0262	0.2711	0.2954	0.6176
the bottom left of this image is	0.1085	0.0293	0.3027	0.3193	0.8302
the bottom left of this picture is	0.1067	0.0365	0.3189	0.3420	0.8320
the bottom left of this image shows	0.1256	0.0515	0.3428	0.3342	0.6091
bottom left of this image: $\_\!$	0.1113	0.0340	0.3561	0.3772	0.7671
bottom left: $\_\!$	0.1093	0.0702	0.3597	0.3770	0.7886

Table 14: Various metrics for prompted generation of base captioning model on cFMNIST. Accuracy and score difference are calculated on the (unperturbed, blur) pair of listeners.

Caption prefix	Accuracy difference	Score difference	Object prevalence	FMNIST keyword prevalence	Color prevalence	Color diversity
the clothing item in this image is $\_\!$	0.0174	0.0491	0.0685	0.1390	0.0801	0.0083
an image of an object: $\_\!$	0.0265	0.0060	0.0743	0.0981	0.0755	0.0103
a black and white image of $\_\!$	0.0493	-0.0071	0.1065	0.1439	1.0000	0.0044
the item in this image is $\_\!$	0.0335	0.0254	0.1147	0.1493	0.0532	0.0083
the object in this image is $\_\!$	0.0361	-0.0008	0.1321	0.1664	0.0791	0.0104
item: $\_\!$	0.0280	0.0433	0.1627	0.2210	0.3257	0.0131
object: $\_\!$	0.0558	-0.0017	0.2274	0.3169	0.3764	0.0141
a picture of $\_\!$	0.0664	0.0044	0.2350	0.3213	0.4122	0.0155
an image of $\_\!$	0.0824	0.0103	0.2646	0.3684	0.4778	0.0157
a black and white image of	0.1170	0.0362	0.3895	0.5977	1.0000	0.0050
the item in this image is	0.0988	0.0449	0.4014	0.5656	0.3498	0.0136
a picture of	0.0997	0.0518	0.4363	0.6556	0.4765	0.0165
an image of	0.1059	0.0466	0.4428	0.6638	0.5181	0.0163
the object in this image is	0.0885	0.0285	0.4438	0.6146	0.4448	0.0155
an image of an object:	0.0967	0.0363	0.4523	0.6343	0.4434	0.0156
object:	0.1183	0.0018	0.4545	0.6533	0.5643	0.0143
item:	0.1015	0.0528	0.4620	0.6731	0.5747	0.0134
the clothing item in this image is	0.1086	0.0546	0.4692	0.8643	0.5157	0.0142

Table 15: Various metrics for prompted generation of base captioning model on cFMNIST. Accuracy and score difference are calculated on the (unperturbed, grayscale) pair of listeners. For reference, the color diversity of the fully specialized speaker in this case is 0.015, which is on par with the best color diversity values across prompts.

Caption prefix	Accuracy difference	Score difference	Color prevalence	Color diversity	Object prevalence	FMNIST keyword prevalence
color color color: „	0.0180	0.0264	0.1421	0.0055	0.0330	0.0433
the colors in this image are „	0.0139	0.0341	0.1697	0.0082	0.0040	0.0045
a picture with the color „	0.0584	0.0410	0.2206	0.0139	0.2622	0.3536
a picture with the color „	0.0851	0.0507	0.2745	0.0144	0.3423	0.4613
a picture with color „	0.0347	0.0395	0.2891	0.0141	0.1583	0.2319
color: „	0.0507	0.0320	0.3000	0.0060	0.0387	0.0501
the color of this image is „	0.0318	0.0341	0.3648	0.0145	0.0058	0.0064
an image with the color „	0.0782	0.0302	0.3862	0.0153	0.2468	0.3197
an image with the color „	0.0952	0.0553	0.3905	0.0153	0.2743	0.3579
an image with color „	0.0268	0.0308	0.4225	0.0136	0.1661	0.2413
the colors in this image are	0.0587	0.0511	0.4336	0.0063	0.0490	0.0547
a picture with color	0.0452	0.0398	0.4396	0.0111	0.1848	0.2455
colors in image: „	0.0097	0.0397	0.5018	0.0020	0.0002	0.0004
color color color: „	0.0764	0.0572	0.5512	0.0033	0.1996	0.2535
an image colored „	0.0919	0.0426	0.5901	0.0123	0.1259	0.1593
an image with color	0.0504	0.0410	0.6026	0.0112	0.1863	0.2555
an image colored	0.1303	0.0631	0.8600	0.0101	0.0200	0.0271
the color of this image is	0.1735	0.0667	0.9010	0.0198	0.0002	0.0002
color: „	0.1180	0.0583	0.9705	0.0040	0.1110	0.1354
colors in image: „	0.0477	0.0428	0.9994	0.0012	0.0000	0.0000

Table 16: Various metrics for prompted generation of base captioning model on COCO. Accuracy and score difference are calculated on the (unperturbed, grayscale) pair of listeners. For reference, the color diversity of the fully specialized speaker in this case is 0.033, which is on par with the best color diversity values across prompts.

Caption prefix	Accuracy difference	Score difference	Color prevalence	Color diversity
an image colored	0.0016	0.0253	0.0109	0.0110
a picture with the color	0.0055	0.0136	0.0117	0.0144
a picture with color „	-0.0039	0.0023	0.0172	0.0155
the color of this image is „	-0.0094	-0.0071	0.0195	0.0180
color color color: „	-0.0086	0.0045	0.0219	0.0172
an image colored „	-0.0195	0.0166	0.0242	0.0190
an image with color „	-0.0141	-0.0010	0.0242	0.0182
the colors in this image are	-0.0008	0.0131	0.0312	0.0171
colors in image: „	-0.0055	-0.0023	0.0344	0.0128
an image with the color	0.0000	0.0115	0.0375	0.0221
a picture with the color „	0.0031	0.0040	0.0383	0.0223
color: „	0.0078	0.0079	0.0508	0.0224
an image with the color „	0.0055	-0.0021	0.0586	0.0261
color color color: „	-0.0016	0.0071	0.0906	0.0278
a picture with color	-0.0016	0.0053	0.0914	0.0165
the colors in this image are „	-0.0156	0.0091	0.1172	0.0317
the color of this image is	-0.0102	0.0012	0.1352	0.0362
an image with color	0.0008	0.0029	0.1477	0.0167
colors in image:	0.0188	0.0052	0.3352	0.0179
color: „	0.0125	0.0047	0.5406	0.0304

## H Sample hands across experimental conditions

In this section we present representative Dixit hands with the corresponding captions produced by the specialized speaker (after training) in our four main conditions, along with the corresponding distractors, listener match scores and rewards. We chose the speaker seed which achieved the highest overall score in that condition (though results were generally comparable across seeds), and to avoid cherry-picking examples we show the first four evaluation hands (from our randomly ordered evaluation) that contained a distinct target object (for example, in the crop condition, the first four that had a distinct object category in the bottom left).

We present samples for the different transformation conditions in Figures 6-9. Overall, the speaker adapts to the target difference between the listeners, and exhibits relatively mild language drift—the utterances generally stay grounded and human-interpretable, but in some cases exhibit some repetition or odd grammar. Some potential methods for further reducing these issues are noted above.

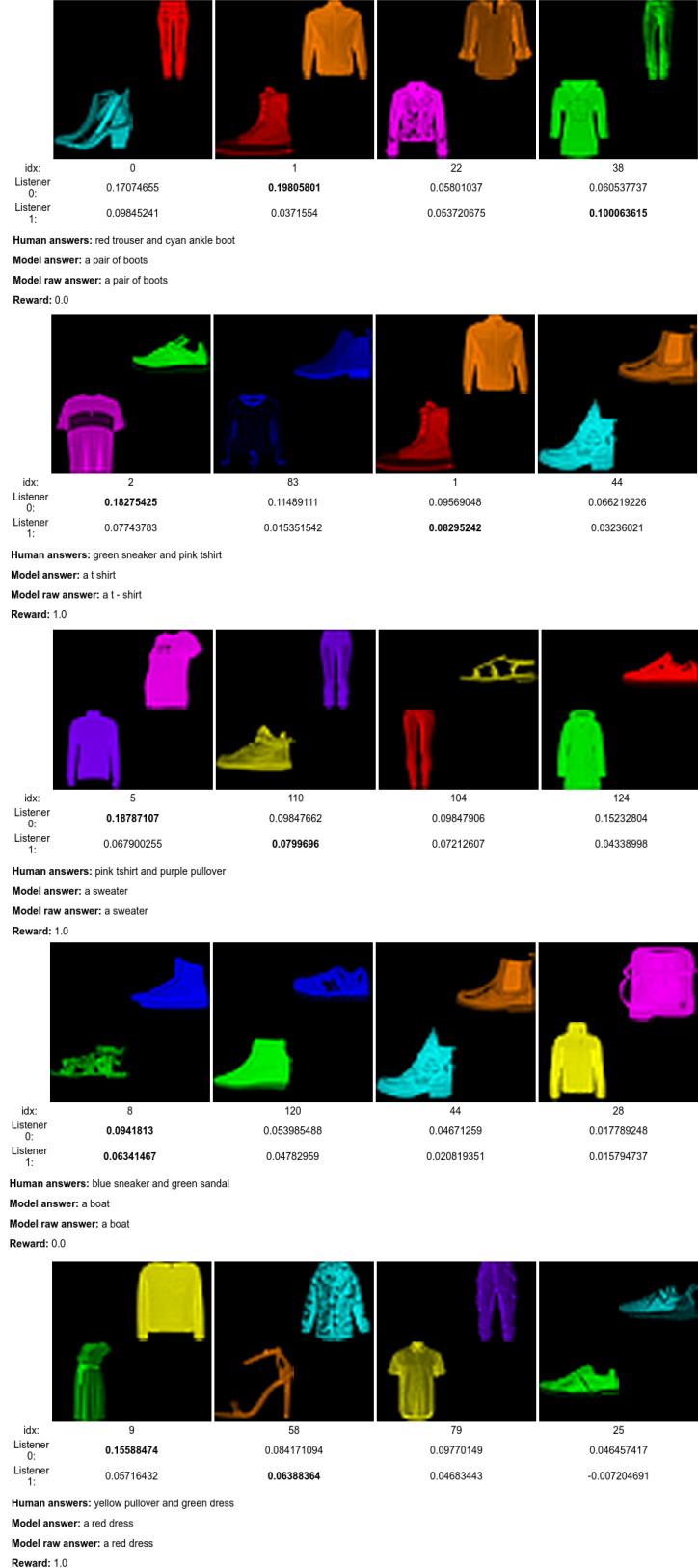


Figure 6: Speaker samples and listener results on tcFMNIST, after the speaker has specialized to the second listener having the crop transformation. The language generally specifies the bottom left object, with a grounding failure in the second-to-last case (or perhaps a reference to boat sandals?).

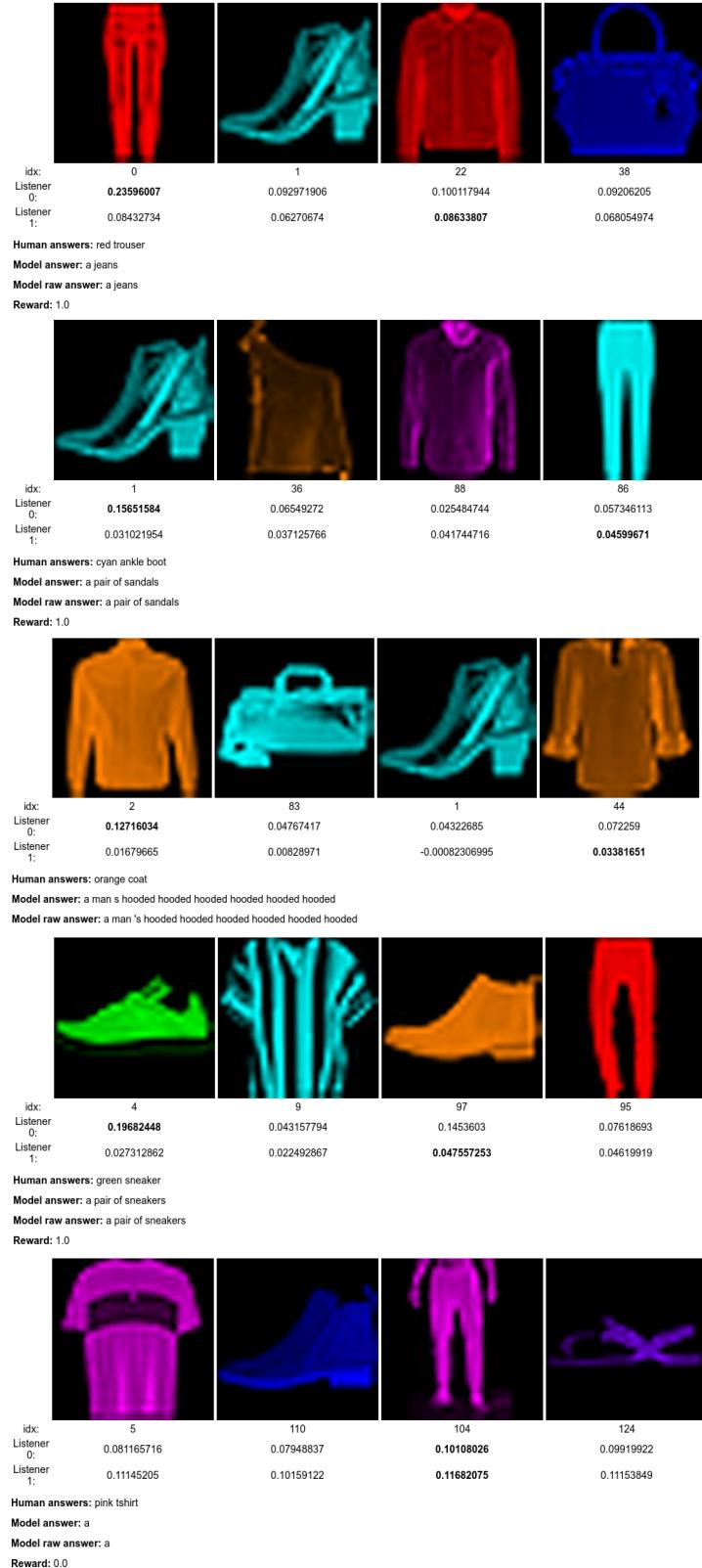


Figure 7: Speaker samples and listener results on cFMNIST, after the speaker has specialized to the second listener having the blur transformation. The speaker generally ignores colors and names the object as intended, but with moderate language degradation in some cases.

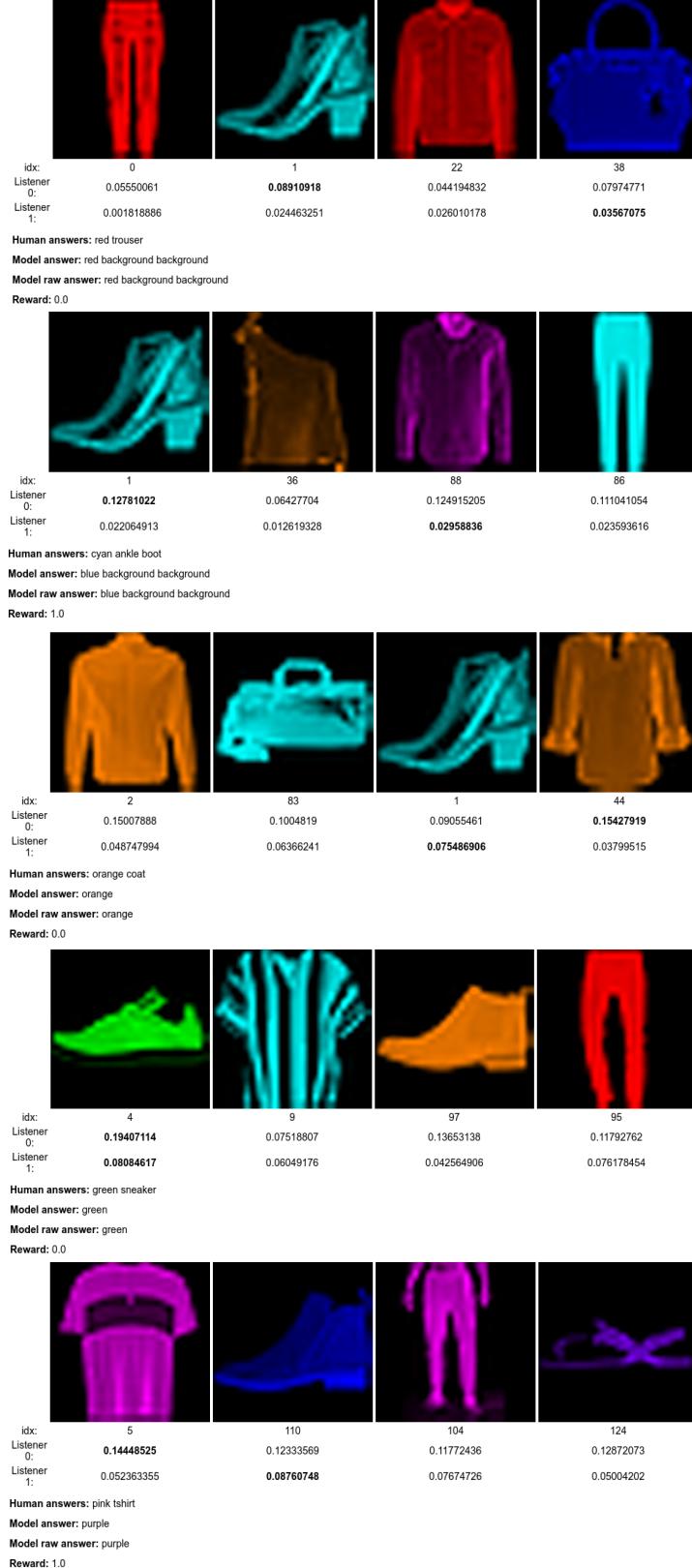


Figure 8: Speaker samples and listener results on cFMNIST, after the speaker has specialized to the second listener having the grayscale transformation. The speaker consistently names the correct color, though it occasionally also repeats “background”.

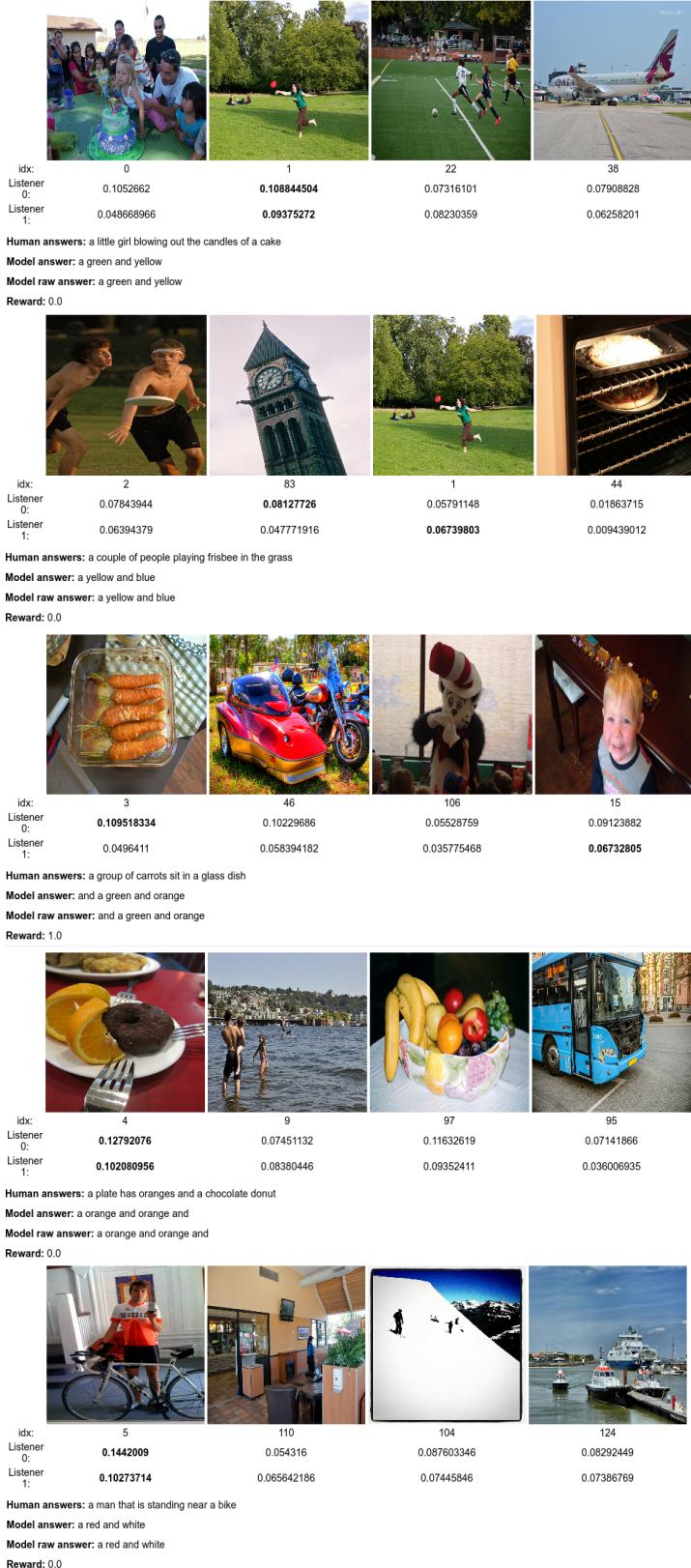


Figure 9: Speaker samples and listener results on COCO, after the speaker has specialized (on COCO) to the second listener having the grayscale transformation. The speaker generally names one or two plausible colors for the images; but these are less discriminative in COCO than in the above results. There is also some minor language degradation in some cases (e.g. “and” at the end of a caption).