# CoGen: Learning from Feedback with Coupled Comprehension and Generation

**Mustafa Omer Gul and Yoav Artzi**
Department of Computer Science and Cornell Tech, Cornell University
{momergul, yoav}@cs.cornell.edu

## Abstract

Systems with both language comprehension and generation capabilities can benefit from the tight connection between the two. This work studies coupling comprehension and generation with focus on continually learning from interaction with users. We propose techniques to tightly integrate the two capabilities for both learning and inference. We situate our studies in two-player reference games, and deploy various models for thousands of interactions with human users, while learning from interaction feedback signals. We show dramatic improvements in performance over time, with comprehension-generation coupling leading to performance improvements up to 26% in absolute terms and up to 17% higher accuracies compared to a non-coupled system. Our analysis also shows coupling has substantial qualitative impact on the system's language, making it significantly more human-like.

## 1 Introduction

Language comprehension and generation are closely related processes. Indeed, observations such as the ability to finish incomplete partner utterances in dialogue (Clark and Wilkes-Gibbs, 1986; Howes et al., 2011), as well as neuroscientific evidence (Paus et al., 1996; Opitz et al., 2003; Menenti et al., 2011) have led to integrated accounts of comprehension and generation in cognitive science (Pickering and Garrod, 2013; Pickering and Gambi, 2018), where processes related to generation are active during comprehension and vice versa. This suggests a potential for coupling the two in computational systems, and creating a virtuous cycle, where the improvement of one capability drives learning and performance in the other. This is particularly compelling in systems that continually learn and improve through interaction with users, where the dynamics between the two capabilities play out over time.

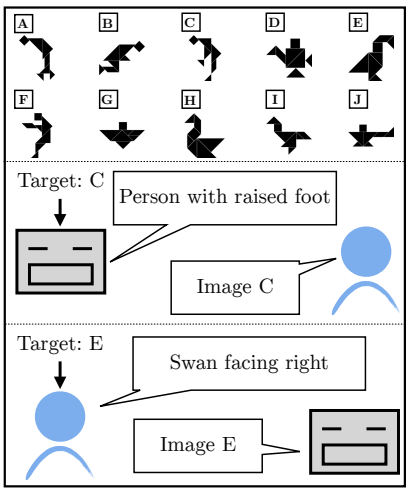


Figure 1: Illustration of our reference game interaction scenario involving a speaker and listener. Each game includes a single turn. Speakers are assigned a target image and write a description such that their partner can guess the image from the description. The game succeeds if the listener guesses correctly. We deploy our models (gray bot) as speaker to interact with human listeners (top) or vice versa (bottom).

We study the dynamics of this coupling in a continual learning[1] setting, where trends in learning and behavior can be observed over time. We design an interaction scenario where models can take both listener (comprehension) and speaker (generation) roles, and receive feedback while interacting with human partners. We couple comprehension and generation through several mechanisms, and observe the impact this coupling has on the long-term dynamics of performance and language.

We instantiate comprehension and generation as the listener and speaker roles of a two-player reference game (Krauss and Weinheimer, 1964; Clark and Wilkes-Gibbs, 1986) involving abstract visual stimuli (Ji et al., 2022), which remain challenging

[1] *Continual learning* is at times used to describe scenarios where models are adapted to new tasks. We use it in the sense of improving a model on its original task over time.

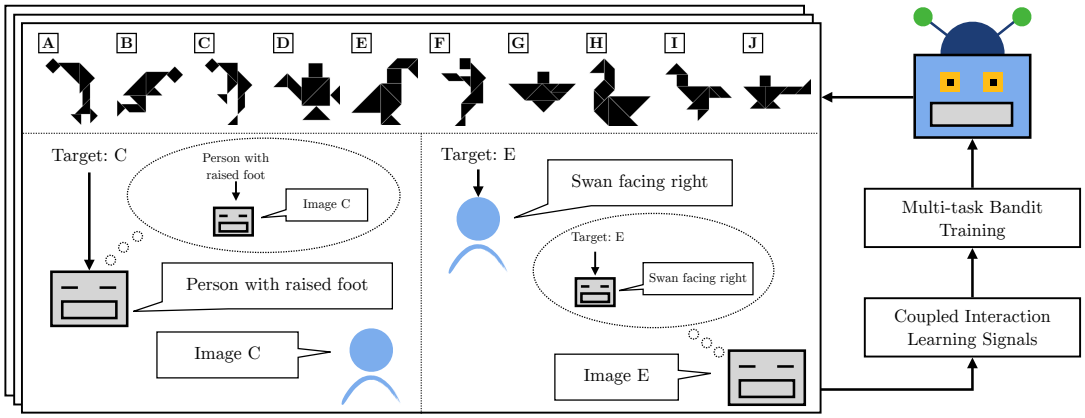


Figure 2: Illustration of our continual learning scenario with coupled comprehension and generation. The process alternates between interactions with human partners in a reference game, and training using learning signals from the interactions. The model performs both the generation (left) and comprehension (right) tasks, while jointly reasoning over the other role (thought bubbles). Training leverages feedback for the role the model performs as well as the opposing role. Following each round of training, we re-deploy the updated model and repeat the process.

for state-of-the-art vision-language models (Figure 1). We deploy a single model that can take both roles. The process alternates between the model interacting with human partners, and training to improve both comprehension and generation capabilities based on feedback from the interactions.

We couple comprehension and generation through two strategies: (a) at inference-time via a joint inference process that incorporates the opposing role, and (b) at training time by generating examples and rewards for each role from feedback on performance in that role as well as the opposing role. Figure 2 illustrates the deployment and coupling mechanisms. The combination of these strategies creates a virtuous cycle that evolves over time, as the system continually trains and improves. As one capability improves (e.g., comprehension), the model's performance on the opposing capability (e.g., generation) also improves via the joint inference procedure. This, in turn, leads to better interactions, and the feedback the model receives changes as its capabilities advance and its failure modes change. The coupling of feedback signals via the training data qualitatively changes the training beyond a simple increase in the amount of data. Whereas a generation system that trains on feedback is only exposed to its own language, the coupled system is continually exposed to a stream of new human language. This can enable the system to expand its generation abilities and make the language more similar to humans, beyond simply refining it to maximize interaction performance.

We conduct extensive experiments, concurrently deploying a baseline and multiple model variants for thousands of interactions with human partners in a controlled study. Our focus is to observe both performance and language trends over time. Our coupled approach shows dramatic and fast performance gains, overall improving by 19.48% for comprehension and 26.07% for generation, in absolute terms. At the conclusion of our deployment, the coupled approach outperforms the non-coupled baseline by 14.80% for comprehension and 17.10% for generation. Furthermore, coupling results in greater data efficiency, with the full system still outperforming this baseline with less than one-third the number of human interactions. We observe coupling dramatically influences the generated language, with the coupled approach exhibiting a larger effective vocabulary and greater alignment with human language according to both linguistic measures such as utterance length and automated metrics such as MAUVE (Pillutla et al., 2021). Our code, data, and experiment logs are available at https://github.com/lil-lab/cogen.

## 2 Interaction Scenario and Overview

We study the coupling of comprehension and generation by training and deploying an agent that interacts with human users, and continually learns from these interactions. This allows us to observe how the interplay between comprehension and generation evolves over time, and what the long-term effects of coupling the two processes are.

**Interaction Scenario** We use a reference game as our interaction scenario (Figure 1). Each game involves two players: a speaker and listener. Both

participants are presented a set of abstract tangram images as context $\mathcal{I} = \{I_1, \ldots, I_N\}$. Each participant observes the images in a different order. The speaker is given a target $I_t \in \mathcal{I}$, and generates an utterance $u$, with the goal of allowing the listener to pick the target $I_t$ from the set of the images. The listener then makes a choice. An interaction succeeds if the listener picks the intended target.

Reference games have been extensively used in research, including in NLP (e.g., Andreas and Klein, 2016; Ji et al., 2022) and cognitive science (e.g., Krauss and Weinheimer, 1964; Rosenberg and Cohen, 1964; Clark and Wilkes-Gibbs, 1986; Hawkins et al., 2023), and provide a balance between complexity and research feasibility: (a) the interaction includes both generation and comprehension; (b) they are relatively accessible for crowdsourcing workers; (c) they are well scoped so learning is feasible without excessive data requirements; and (d) success is easy to measure. The tangram shapes we use have been shown to elicit rich linguistic behavior, both from listeners and speakers (Schober and Clark, 1989; Ji et al., 2022). They also remain challenging for contemporary models. Our models' initial performance is at least 33.1% below human accuracy (Section 6).

**Deployment** We deploy our model to interact with humans in rounds. Each round includes a predetermined number of interactions, each with the model taking one of the roles (speaker or listener) and the human participant taking the other. We derive feedback signals from the interactions, construct training examples, and train our model. Following training, we re-deploy for the next round.

**Inference and Learning** We use IDEFICS2-8B (Laurençon et al., 2024) as our model, an auto-regressive LLM that can also take images as part of its input. The model is parameterized by $\theta$. As a speaker (generation), the model computes a probability distribution $P_s(u|\mathcal{I}, t; \theta)$ over descriptions $u$ of the target $t \in \{1, \ldots, N\}$ in the given context $\mathcal{I}$. As a listener (comprehension), it computes a distribution $P_l(t|\mathcal{I}, u; \theta)$ over target $t$ selections, given the context $\mathcal{I}$ and a description utterance $u$. Both utterances and target selections are generated via a conventional auto-regressive process. Following each deployment round, we train the model using all the feedback data collected so far by treating the feedback as rewards for contextual bandit learning.

**Evaluation** Our main evaluation is conducted through interaction with humans, where each round forms the evaluation of the model so far. We evaluate the comprehension performance of the model from its target selection accuracy as a listener. We evaluate generation as the accuracy of the human listener in selecting the target given the model's generated description when in the speaker role. We also study the linguistic trends of the model's generations over time to better understand the dynamics created by coupling comprehension and generation. This includes analyzing its similarity to human language and its linguistic properties.

## 3 Continual Learning

We combine the continual learning approaches of Kojima et al. (2021) (generation) and Suhr and Artzi (2024) (comprehension). Both approaches map feedback to rewards, treat learning as a contextual bandit problem, and use REINFORCE (Williams, 1992), a relatively simple policy gradient algorithm. We adopt these design choices. A key difference of our process is that we combine the comprehension and generation objectives to train a single model.

Deployment and learning are interleaved. Each round $\rho$ starts with deploying the model parameterized by $\theta_\rho$ to collect interactions with humans. We record feedback signals from these interactions. Upon collecting a set of interactions, we re-train the model given all data collected so far to estimate new parameters $\theta_{\rho+1}$. The model is then deployed for the next round, and the process continues.

### 3.1 Feedback Collection

Feedback collection is part of the model interacting with human partners (i.e., the system deployment), and differs depending on the model's role. As the listener, the model is given context $\mathcal{I}$ and a human-generated utterance $u$ and predicts the index of the target image $\hat{t} = \arg\max_t P_l(t|\mathcal{I}, u; \theta_\rho)$. The game then indicates if the selection was correct or not, and terminates. We treat this indication as feedback,[2] and directly map it to a binary reward to create a comprehension datapoint: $(\mathcal{I}, u, \hat{t}, r)$, with $r = 1$ upon game success and $r = -1$ otherwise. Likewise, as the speaker, the model samples an utterance $\hat{u} \sim P_s(u|\mathcal{I}, t; \theta_\rho)$ given context $\mathcal{I}$ and tar-

[2] In single-turn reference games, such as in our scenario, task success and feedback are the same, so we do not solicit explicit feedback. However, our approach is designed to be applicable to settings where feedback and task success do not collapse to be the same, such as the setting considered by Kojima et al. (2021) and Suhr and Artzi (2024).

get image index $t$. The game indication of success provides the feedback, resulting in a generation datapoint: $(\mathcal{I}, \hat{u}, t, r)$ with $r \in \{-1, 1\}$ accordingly. Each round results in two datasets: $\mathcal{D}_{l,\rho}$ and $\mathcal{D}_{s,\rho}$ for comprehension and generation. In both, datapoints constitute model output produced during interaction, and a reward for it. This is in contrast to supervised learning, where datapoints include output annotations, or human feedback as used in RLHF, where datapoints are pairwise preferences drawn from external annotators.

### 3.2 Learning

We estimate the next round's model parameters $\theta_{\rho+1}$ by re-training from the initial weights (i.e., the original IDEFICS2 weights). The comprehension training dataset is a union of all collected feedback data so far $\mathcal{D}_{l,\leq\rho} = \bigcup_{i=1}^{\rho} D_{l,i}$. The production task dataset $\mathcal{D}_{s,\leq\rho}$ is similarly defined.

We frame learning as a contextual bandit problem with a multi-task additive objective combining the comprehension and generation components. We optimize with a REINFORCE-style policy gradient algorithm (Williams, 1992). This choice follows prior work (Kojima et al., 2021; Suhr and Artzi, 2024), and is motivated by the simplicity of REINFORCE, critical in a setting where humans are part of the iterative learning process.[3] The gradient for a comprehension example $(\mathcal{I}, u, \hat{t}, r) \sim \mathcal{D}_{l,\leq\rho}$ collected at round $m$ is:

$$\Delta_l = c_l r \nabla \log P_l(\hat{t}|\mathcal{I}, u; \theta) \quad , \tag{1}$$

where $c_l$ is the cased inverse propensity score (IPS) coefficient introduced by Kojima et al. (2021) to mitigate the effect of negative examples (i.e., $r = -1$) allowing for unbounded loss:

$$c_l = \begin{cases} \frac{P_l(\hat{t}|\mathcal{I},u;\theta)}{P_l(\hat{t}|\mathcal{I},u;\theta_m)} & \text{if } r = -1 \\ 1 & \text{else} \end{cases} \quad , \tag{2}$$

where $P_l(\hat{t}|\mathcal{I}, u; \theta_m)$ is the probability of the target $\hat{t}$ when it was sampled during the interaction at round $m$. Without this coefficient, negative examples (i.e., $r = -1$) can dominate the loss and destabilize learning as their probabilities decrease, because $\lim_{P_l(\cdot)\to 0} \log P_l(\cdot) = -\infty$. The coefficient $c_l$ decreases the importance of such examples as their probability decreases. The gradient $\Delta_s$ for generation datapoints $(\mathcal{I}, \hat{u}, t, r)$ is identical, except using the generation distribution $P_s(\hat{u}|\mathcal{I}, t; \theta)$.

[3] More generally, Ahmadian et al. (2024) recently showed REINFORCE can match more modern methods, such as PPO.

## 4 Coupling Comprehension and Generation

We couple comprehension and production during both learning and inference. We also use one model for both tasks, creating a coupling at the parameter level, which is common in contemporary methods, partially due to high memory needs.

### 4.1 Learning with Data Sharing

We convert comprehension datapoints to generation datapoints, and vice versa, to fully utilize the data models are exposed to in interactions. For example, consider the case of an agent in the role of a listener. If the speaker partner generates the utterance *the target is a swan facing right* and the listener correctly guesses the target image (as in Figure 2), the listener does not only receive positive feedback for their guess, but also can learn that *a swan facing right* is a valid description for the current context-target pair.

Given datasets for comprehension $\mathcal{D}_{l,\rho}$ and generation $\mathcal{D}_{s,\rho}$ collected at round $\rho$, we expand both:

$$\mathcal{D}_{l,\rho} = \mathcal{D}_{l,\rho} \cup \{(\mathcal{I}, \hat{u}, t, r) \in \mathcal{D}_{s,\rho} \mid r = 1\}, \tag{3}$$

$$\mathcal{D}_{s,\rho} = \mathcal{D}_{s,\rho} \cup \{(\mathcal{I}, u, \hat{t}, r) \in \mathcal{D}_{l,\rho} \mid r = 1\}. \tag{4}$$

We only convert positively labeled feedback ($r = 1$), because we generally find positive rewards to be more reliable. A negative reward for a generated utterance could be because the utterance is incorrect or ambiguous, or the human listener made a mistake. The listener task is essentially classification. Creating a comprehension example with negative reward from such an example indicates to the model the utterance is a valid description for another target. This is a misleading signal, and in early pilot studies we found it not to be helpful, so we only convert examples with positive reward.

An important result of this process is introducing human language into the training data of the speaker model. Generally, if a generating model learns from feedback only (Kojima et al., 2021), it is only exposed to language it has generated. This can lead to its language drifting from human language, even if its accuracy and legibility to human partners increase. Taking advantage of human utterances for the purpose of generation training opens up this closed system. We further discuss this in our results and analysis (Section 6).

### 4.2 Joint Inference

We couple the two distributions $P_l$ and $P_s$ during inference by sampling from one distribution (i.e., $P_l$ in the case of comprehension) and then re-rank with a weighted geometric mean of the two distributions. The weight controlling the geometric mean is a hyper-parameter: $\lambda_s$ for generation and $\lambda_l$ for comprehension. In the case of comprehension, the joint probability distribution is:

$$P_l^j(t|\mathcal{I}, u; \theta) = \frac{P_l(t|\mathcal{I}, u; \theta)^{\lambda_l} P_s(u|\mathcal{I}, t; \theta)^{1-\lambda_l}}{\sum_{t'=1}^{N} P_l(t'|\mathcal{I}, u; \theta)^{\lambda_l} P_s(u|\mathcal{I}, t'; \theta)^{1-\lambda_l}}, \quad (5)$$

where $N$ is the number of targets. The joint generation distribution $P_s^j(u|\mathcal{I}, t; \theta)$ is defined in a similar fashion, but with the $\lambda_s$ hyperparameter. Enumerating all possible utterances for the normalization of the joint generation distribution is intractable, so we sample $k$ utterances from $P_s(u|\mathcal{I}, t; \theta)$ and sum over them to compute the normalization. In the case of comprehension, we can compute the joint distribution exactly because the number of outputs is small (i.e., 10 targets). However, if the number of targets was intractably large, the same approximation could also be performed for comprehension. In practice, we observe the multiplicative generation distribution to skew inference heavily towards short utterances when doing joint inference, and find $\lambda_s = 0$ to be the best combination for the joint generation distribution (Section 5). Although this eliminates the term $P_s$ from the joint probability, $P_s$ is still influential as the source of samples.

This joint formulation is similar to a rational speech act model (RSA; Goodman and Frank, 2016) with a single level of recursion. RSA is a model of pragmatic reasoning, and has been evaluated extensively in reference games (Cohn-Gordon et al., 2018; McDowell and Goodman, 2019). We analyze this property for our speaker model in Section 6.2. Our approximation of the joint speaker distribution is inspired by similar approaches that were applied to RSA (Fried et al., 2018a).

## 5 Experimental Setup

**Game Construction** We construct reference game contexts using the KILOGRAM dataset (Ji et al., 2022) of 1,016 abstract tangram shapes. Each context comprises 10 images drawn from this dataset. We use a CLIP model (Radford et al., 2021) finetuned on KILOGRAM annotations from Ji et al. (2022) to ensure visual similarity between images in each context and increase task difficulty. Appendix B provides further details.

**Model and Initialization** We fine-tune the instruction-tuned IDEFICS2-8B (Laurençon et al., 2024). The tasks are delineated via prompting. Training hyperparameters are kept fixed throughout different continual learning rounds and system variants. For systems with joint inference, we set $\lambda_L = 0.5$ and $\lambda_S = 0$. Appendix A details prompt design, hyperparameters, and training. Before the first round of interactions, we initialize the model by fine-tuning IDEFICS2 with a small set of 104 successful human-human games. We also add this data to the later rounds of re-training, by assigning all these examples a reward of 1. We use 280 successful human-human games as a validation set for model selection throughout our experiments.

**System Variants** We refer to our proposed system coupling comprehension and generation with joint inference and data sharing as FULL. We compare against three other systems: ablations without data sharing (NO-DS; Section 4.1) or joint inference (NO-JI; Section 4.2), and a baseline that uses neither (BASELINE). We additionally collect human-human interaction data (HUMAN) to contextualize performance over time relative to human performance. We also use this human-human data for language analysis.

**Deployment** We conduct four rounds of deployment, including interactions with human partners and learning. All interactions for each round are collected concurrently in a randomized experiment. We collect an equal number of interactions for the speaker and listener roles for each system and round. We collect 2,000 interactions for each role for each system in the first round, and increase the number by 500 each round, as the marginal benefit of more examples decreases as the data grows.[4] Because data sharing is not applicable for the first round, the FULL and NO-DS, and the NO-JI and BASELINE systems are identical on the first round. We deploy our systems to interact with human workers on MTurk, at a total cost of $12,980USD. Appendix E provides crowdsourcing details.

**Evaluation** At each round, we evaluate comprehension performance from interactions in the listener role using the target selection accuracy. We

[4] Comprehension and generation performance are identical for the HUMAN system, so we collect half the number of interactions for that system.

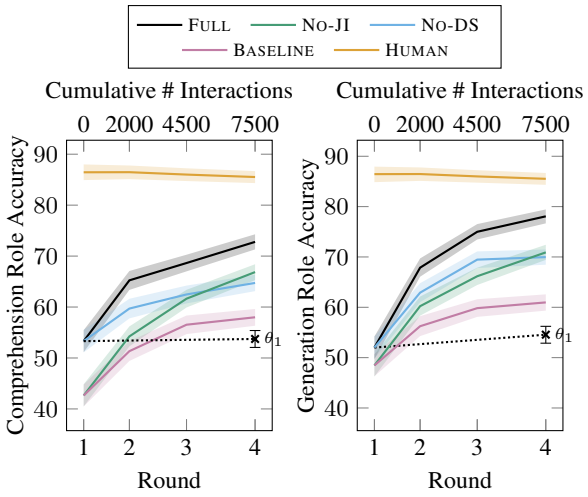


Figure 3: Comprehension and generation performance for system variants across four rounds of deployment, with 95% confidence intervals.[5] The top $x$-axis indicates the total number of interactions collected for a role up to the deployment round. Coupling comprehension and generation leads to FULL outperforming all ablations throughout.

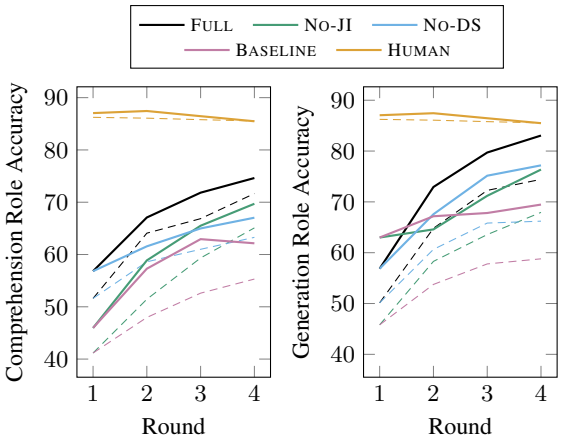


Figure 4: Model comprehension and generation accuracy when the speaker utterance includes (⋰) and does not include (／) words for spatial reasoning.

evaluate a system's generation performance (i.e., as a speaker) as the accuracy of the human interlocutor's target selections. For HUMAN, comprehension and generation performance are identical.

## 6 Results and Analysis

We focus on two broad questions: (a) does coupling influence the rate of improvement on task performance (Section 6.1) and (b) does it lead to quantifiable differences in the generated language over time (Section 6.2). Overall, we find the answer to both questions is positive, with strong effects.

### 6.1 Performance Analysis

Figure 3 shows model performance over time. All systems show dramatic improvement in performance for both comprehension and generation. Immediately, we observe significant effect from joint inference, with FULL and NO-DS outperforming NO-JI and BASELINE on the first round (53.31% vs. 42.64% comprehension, 52.00% vs. 48.45% generation). FULL achieves the highest performance at the end of the study, with comprehension improving 53.31→72.79% (19.48% absolute improvement) and generation 52.00→78.07% (26.07% improvement). For generation, FULL shows the biggest performance delta, even though it starts with already higher performance compared to variants without joint inference. With comprehension, NO-JI (42.64→66.86%) shows the biggest delta (24.22%). Coupling dramatically increases learning sample efficiency: FULL at the second round already performs better than BASELINE at the end of study, even though it trained on less than one third of the data BASELINE has seen at the end.

Overall, the gap in performance between FULL and BASELINE only increases over time. For comprehension, the gap widens 10.67→14.80%, but it is much more dramatic for generation with 3.55→17.10%. Both coupling strategies play a role in this widening gap in performance, but between the two strategies the relation changes over time. Although NO-DS starts with higher performance than NO-JI, they are essentially equivalent at the end, with NO-JI showing a trend of outperforming NO-DS. This may be because NO-JI is exposed to more data from the opposing role with data sharing, compensating for the lack of joint inference.[6]

User adaptation is an important potential confounder, potentially explaining any improvements in system performance. During the final round, we deploy the initial FULL model in a concurrent randomized deployment. We observe that human adaptation cannot explain model improvement, seeing very limited improvement due to adaptation: 0.42% and 2.56% for comprehension and generation (cross and dashed curve in Figure 3).

During deployment, a recurring complaint from workers was about the models' inconsistent spatial reasoning, echoing recent evaluations of vision-

[5] Confidence intervals are computed using bootstrap sampling, where $n = 10{,}000$.

[6] Figure 8 in Appendix D.2 depicts data sharing's impact on training set size over time.

language models (Kamath et al., 2023; Tong et al., 2024a,b). We identified games where utterances involve a word relating to spatial reasoning.[7] Figure 4 shows a breakdown of performance trends to games that contain spatial reasoning utterances and games that do not. We see a clear difference between the two sets. Although models improve on utterances that contain spatial reasoning, they perform worse on them throughout. During the final round, we observe that FULL's performance nears that of humans for generation when not using words for spatial reasoning.

Coupling demonstrates a very strong effect, both on performance and language trends. Balancing the utility of further rounds versus the high cost of each round, we ended the deployment after four rounds. Appendix D.1 discusses this decision, and provides an extrapolation of performance for one more round, showing a continuation of the observed trends.

### 6.2 Language Analysis

We study trends in language use over time. Throughout this section, except the pragmatic reasoning analysis, we eliminate factors that can complicate the analysis by generating new utterances on the same set of context-target pairs per round for all systems. We randomly sample 2,000 context-target pairs from the human-human games for each round, and generate utterances for them with each system using the same inference process as during deployment. Figure 5 plots the observed trends.[8]

We observe a decrease in utterance length for all variants. Humans also show a downward trend in length, likely reflecting the participants becoming experts and therefore more economical in their language. This is a known phenomenon in reference games (Krauss and Weinheimer, 1964; Clark and Wilkes-Gibbs, 1986), and was also observed in other collaborative scenarios (Effenberger et al., 2021). FULL and NO-JI track the human trends best, but generally generate shorter utterances throughout.

The effective vocabulary of all systems, that is the number of unique words generated for the set of context-target pairs, is also decreasing. This has been observed in prior studies for generation systems that are exposed only to their output in continual learning (Kojima et al., 2021). We expected this effect to be less strong or even reversed once the system is exposed to human utterances, either through data sharing or through joint inference with a comprehension model trained on human utterances. The decrease in the vocabulary size is much smaller for the coupled variants, and the smallest for FULL, but it remains present. We also plot, for each round, how many words a model added to the cumulative set of words it generated until that round (third panel). More new words appear for the coupled variants throughout the study. All systems display a significantly less rich vocabulary compared to humans, leaving an important direction for future work.

We use MAUVE (Pillutla et al., 2021), a reference-less generation evaluation metric, to evaluate the similarity of each model's language to human language. For each round and system, we compute the metric between the model- and human-produced utterances for that round. We use GPT2-Large as the embedding model (Radford et al., 2019), similar to Pillutla et al. (2021), and keep the number of clusters fixed at 200. We find coupling avoids the drift from human language the BASELINE displays. The FULL system not only does not stray further from human language, but actually moves closer to it over time. Data sharing is particularly critical, but the combination of joint inference further helps to align the model language with human language.

Finally, we briefly look into whether coupling affects the model's pragmatic reasoning. In reference games, the pragmatic information is the images in the context that are not the target. A speaker that employs pragmatic reasoning well will take into account the other images so to help the speaker make the right selection in the specific context they share (i.e., the speaker will refer to properties of the target that specifically distinguish from the other images). We operationalize this question by measuring the diversity of model descriptions for a specific tangram within different context sets. We use the Shape Naming Divergence (SND) metric, introduced by (Ji et al., 2022) to measure the diversity of human annotations for individual tangrams.[9] Roughly speaking, high SND means high lexical diversity between the descriptions of a specific image. For each system and each round, we generate utterances for every context-target pair observed in all human-human games throughout continual

[7] Appendix C.1 provides the set of words we considered.

[8] For all analysis but MAUVE, utterances are lowercased and tokenized with spaCy (Honnibal et al., 2020).

[9] We describe SND in Appendix C.2.

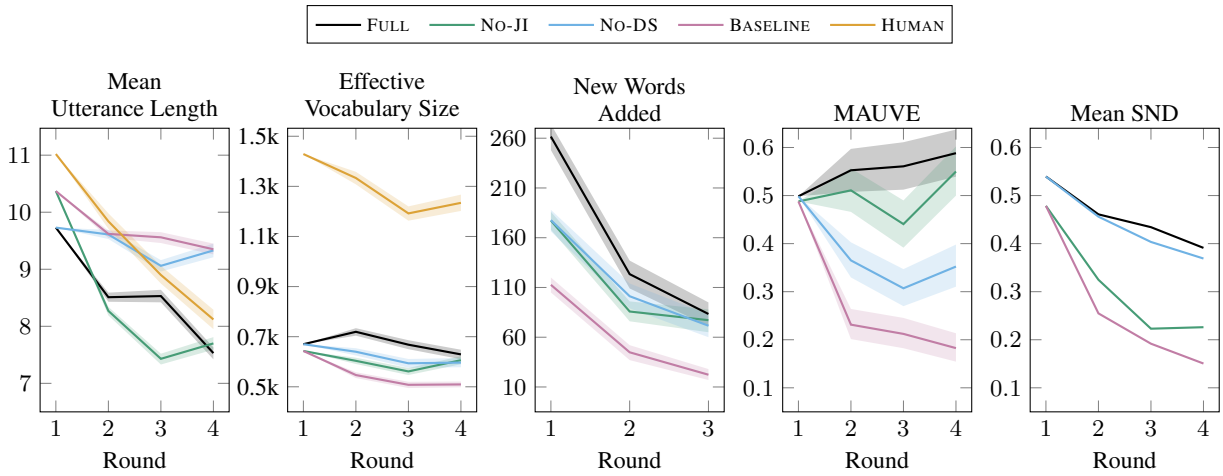


Figure 5: Language analysis plots, with 95% confidence intervals.[11] Trends in utterance length mirror that of humans when using data sharing (FULL and NO-JI). FULL possesses the highest effective vocabulary size and produces the largest number of new words each round. The FULL system additionally shows an increase in MAUVE scores (↑) over time and exhibits the highest SND (↑) throughout.

learning.[10] We get 10.67 utterances per tangram on average. Figure 5 (right pane) shows mean SND across all tangrams for each model and round. Largely, we observe BASELINE's pragmatic ability to collapse over time. Data sharing helps to some degree. While we see a decrease in SND over time even when using joint inference, this type of coupling shows much higher SND values throughout, indicating greater diversity of utterances and hence a greater pragmatic effect. While this effect tracks the vocabulary size trends in practice, it is independent, even if a diverse vocabulary is a necessary, but insufficient condition. That said, this analysis of pragmatic reasoning is rudimentary, and future in-depth analysis is required to identify the exact qualities of this phenomena and how it correlates with system performance.

## 7 Related Work

Our joint inference strategy (Section 4.2) is technically based on approximations (Fried et al., 2018a,b) of the Rational Speech Acts framework (RSA; Goodman and Frank, 2016; Yuan et al., 2018), which frames pragmatic reasoning as a recursive process between listener and speaker models. RSA has been studied extensively with the focus of developing models that reason pragmatically (e.g., Monroe et al., 2017; Andreas and Klein, 2016), including through incorporation in learning (McDowell and Goodman, 2019) and inference (White et al., 2020). We use it for different aims, as one of two strategies to couple comprehension and generation. Liu et al. (2023) studied the incorporation of joint inference for generation learning, which is a component of our study, with a static model listener. In contrast, we study learning dynamics for both comprehension and generation, evaluate data sharing as an additional coupling mechanism, and deploy for continual learning with humans, who constitute non-static partners.

Continually learning from interactions with human users has been studied in the context of instruction generation (Kojima et al., 2021) and following (Suhr and Artzi, 2024), question answering (Gao et al., 2023), and ad-hoc adaptation (Hawkins et al., 2020). In our work, continual learning enables us to study long-term dynamics that arise from coupling comprehension and generation. Our continual learning setup is different from the Reinforcement Learning from Human Feedback framework (RLHF; Ziegler et al., 2019) in relying on binary signals derived from interactions with users, while RLHF requires external annotators that compare output pairs.

The reference game scenario has been extensively used in cognitive studies as a prototypical, but simple interaction design (Rosenberg and Cohen, 1964; Krauss and Weinheimer, 1964). It has been used to study convention formation at dyadic (Clark and Wilkes-Gibbs, 1986; Wilkes-Gibbs and Clark, 1992) and population-levels (Hawkins et al., 2023), and demonstrate com-

[10] We cannot compute SND for human participants because of insufficient data per round.

[11] Confidence intervals are computed over $n = 10{,}000$ random samples of 2,000 context-target pairs for each round.

putational theories of pragmatic reasoning (Goodman and Frank, 2016; Cohn-Gordon et al., 2019), among other behaviors. It has also been used to develop computational methods, such as to evaluate contrastive captioning (Vedantam et al., 2017; Ou et al., 2023) and abstract reasoning of vision-language models (Ji et al., 2022). The tangram images we use, abstract shapes composed of the same set of seven primitives, likewise have extensive use as stimuli in cognitive science (Clark and Wilkes-Gibbs, 1986; Schober and Clark, 1989; Horton and Gerrig, 2002). They also remain challenging for contemporary models (Ji et al., 2022), making them well suited to demonstrate model improvement.

## 8 Conclusion

We study the dynamics of coupling language comprehension and generation at inference- and training-time through a continual learning setting where an agent learns from interactions with humans. Coupling has significant impact over time, leading to improved agent performance, sample efficiency, and similarity to human language.

Our work points to multiple directions for future work, including coupling the processes through the training objective in addition to data at training-time, developing more efficient alternatives to sampling utterances during joint inference for generation, and the study of alternative interaction scenarios, including multi-turn settings where dynamics between comprehension and generation can affect an interaction throughout its duration. Scaling up our approach and experimental setting to a real-world deployment featuring a wider range of tasks and a broader set of feedback signals, such as natural language feedback, constitutes a particularly important direction.

## Limitations

Our work does not touch on an important factor in deployed systems: the addition of new participants into the system. To simplify the crowdsourcing setup, we keep the set of workers fixed during our experiments. This does not allow us to observe the effect of new participants joining the population and the impact of the data they create interacting with our agents. This is an important direction for future work. While our methods are not specifically designed for English, our study is only done in English. We restrict the language to English and recruit workers from English-majority locales only. This qualifies our findings, both with regard to the language choice and the impact of the culture of the participants. These are also important variables for future studies.

Unlike how RL is usually studied in the research community, our continual learning process involves humans in the loop. This entails restrictions in terms of time and cost. We opt for simplicity and choose to train models with a REINFORCE-style policy gradient algorithm (Williams, 1992) and retrain models from scratch on the cumulative set of collected data with each round of continual learning. A more extensive (and costly) search over methods might impact results. We leave the study of more complex RL algorithms, such as PPO (Schulman et al., 2017), as well as different strategies for incorporating data from previous rounds to future work.

We invested significant effort and resources in running our study for a significant amount of interactions and rounds. While we show consistent trends, it is hard to predict trends at much larger scale (e.g., thousands of rounds or millions of interactions). This is beyond the resource available for this research. That said, even if trends change dramatically with such a long horizon, our approach remains useful for faster learning (i.e., reduce regret) in the early life of the system.

## Ethical Considerations

Our work studies how the coupling of comprehension and generation affects the dynamics of performance and model language. Through coupling at training time, our model trains on both its own generations alongside generations its human partners produced at interaction time. A naive implementation of this strategy during real-world deployment risks aligning model behavior with the biases of its human interlocutors at best and exposing the model to adversarial actors at worst. Appropriate guardrails or further research for selecting when to apply data sharing should be implemented before deployment is considered to ensure safety.

## Acknowledgements

This research was supported by ARO W911NF21-1-0106, NSF under grant No. 1750499, a gift from Open Philanthropy, and a gift from Apple. We thank the Cornell NLP group and Siddhartha Datta for discussion and comments; Vivian Chen, Gloria Geng, and Anne Wu for feedback on the crowd-

sourcing pipeline; Vivian Chen and Gloria Geng for sharing code for data visualization; Ron Eliav and Anya Ji for allowing us to build on top of their reference game interface; and the crowdsourcing workers for their participation.

## References

Arash Ahmadian, Chris Cremer, Matthias Gallé, Marzieh Fadaee, Julia Kreutzer, Olivier Pietquin, Ahmet Üstün, and Sara Hooker. 2024. Back to basics: Revisiting REINFORCE-style optimization for learning from human feedback in LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12248–12267, Bangkok, Thailand. Association for Computational Linguistics.

Abdullah Almaatouq, Joshua Becker, James P Houghton, Nicolas Paton, Duncan J Watts, and Mark E Whiting. 2021. Empirica: a virtual lab for high-throughput macro-level experiments. *Behavior Research Methods*, 53(5):2158–2171.

Jacob Andreas and Dan Klein. 2016. Reasoning about pragmatics with neural listeners and speakers. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1173–1182, Austin, Texas. Association for Computational Linguistics.

Herbert H Clark and Deanna Wilkes-Gibbs. 1986. Referring as a collaborative process. *Cognition*, 22(1):1–39.

Reuben Cohn-Gordon, Noah Goodman, and Christopher Potts. 2018. Pragmatically informative image captioning with character-level inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 439–443, New Orleans, Louisiana. Association for Computational Linguistics.

Reuben Cohn-Gordon, Noah Goodman, and Christopher Potts. 2019. An incremental iterated response model of pragmatics. In *Proceedings of the Society for Computation in Linguistics (SCiL) 2019*, pages 81–90.

Anna Effenberger, Rhia Singh, Eva Yan, Alane Suhr, and Yoav Artzi. 2021. Analysis of language change in collaborative instruction following. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2803–2811, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Daniel Fried, Jacob Andreas, and Dan Klein. 2018a. Unified pragmatic models for generating and following instructions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1951–1963, New Orleans, Louisiana. Association for Computational Linguistics.

Daniel Fried, Ronghang Hu, Volkan Cirik, Anna Rohrbach, Jacob Andreas, Louis-Philippe Morency, Taylor Berg-Kirkpatrick, Kate Saenko, Dan Klein, and Trevor Darrell. 2018b. Speaker-follower models for vision-and-language navigation. *Advances in Neural Information Processing Systems*, 31.

Ge Gao, Hung-Ting Chen, Yoav Artzi, and Eunsol Choi. 2023. Continually improving extractive QA via human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 406–423, Singapore. Association for Computational Linguistics.

Noah D Goodman and Michael C Frank. 2016. Pragmatic language interpretation as probabilistic inference. *Trends in cognitive sciences*, 20(11):818–829.

Robert Hawkins, Minae Kwon, Dorsa Sadigh, and Noah Goodman. 2020. Continual adaptation for efficient machine communication. In *Proceedings of the 24th Conference on Computational Natural Language Learning*, pages 408–419, Online. Association for Computational Linguistics.

Robert D Hawkins, Michael Franke, Michael C Frank, Adele E Goldberg, Kenny Smith, Thomas L Griffiths, and Noah D Goodman. 2023. From partners to populations: A hierarchical bayesian account of coordination and convention. *Psychological Review*, 130(4):977.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

William S Horton and Richard J Gerrig. 2002. Speakers' experiences and audience design: Knowing when and knowing how to adjust utterances to addressees. *Journal of Memory and Language*, 47(4):589–606.

Christine Howes, Matthew Purver, Patrick GT Healey, Gregory J Mills, and Eleni Gregoromichelaki. 2011. On incrementality in dialogue: Evidence from compound contributions. *Dialogue & Discourse*, 2(1):279–311.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Anya Ji, Noriyuki Kojima, Noah Rush, Alane Suhr, Wai Keen Vong, Robert Hawkins, and Yoav Artzi. 2022. Abstract visual reasoning with tangram shapes. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 582–601, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Amita Kamath, Jack Hessel, and Kai-Wei Chang. 2023. What's "up" with vision-language models? investigating their struggle with spatial reasoning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9161–9175, Singapore. Association for Computational Linguistics.

Noriyuki Kojima, Alane Suhr, and Yoav Artzi. 2021. Continual learning for grounded instruction generation by observing human following behavior. *Transactions of the Association for Computational Linguistics*, 9:1303–1319.

Robert M Krauss and Sidney Weinheimer. 1964. Changes in reference phrases as a function of frequency of usage in social interaction: A preliminary study. *Psychonomic Science*, 1:113–114.

Hugo Laurençon, Léo Tronchon, Matthieu Cord, and Victor Sanh. 2024. What matters when building vision-language models? *arXiv preprint arXiv:2405.02246*.

Andy Liu, Hao Zhu, Emmy Liu, Yonatan Bisk, and Graham Neubig. 2023. Computational language acquisition with theory of mind. In *The Eleventh International Conference on Learning Representations*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *The Seventh International Conference on Learning Representations*.

Bill McDowell and Noah Goodman. 2019. Learning from omission. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 619–628, Florence, Italy. Association for Computational Linguistics.

Laura Menenti, Sarah ME Gierhan, Katrien Segaert, and Peter Hagoort. 2011. Shared language: overlap and segregation of the neuronal infrastructure for speaking and listening revealed by functional mri. *Psychological science*, 22(9):1173–1182.

Will Monroe, Robert X.D. Hawkins, Noah D. Goodman, and Christopher Potts. 2017. Colors in context: A pragmatic neural model for grounded language understanding. *Transactions of the Association for Computational Linguistics*, 5:325–338.

Philipp Moritz, Robert Nishihara, Stephanie Wang, Alexey Tumanov, Richard Liaw, Eric Liang, Melih Elibol, Zongheng Yang, William Paul, Michael I Jordan, et al. 2018. Ray: A distributed framework for emerging {AI} applications. In *13th USENIX symposium on operating systems design and implementation (OSDI 18)*, pages 561–577.

B Opitz, K Müller, AD Friederici, et al. 2003. Phonological processing during language production: fmri evidence for a shared production-comprehension network. *Cognitive Brain Research*, 16(2):285–296.

Jiefu Ou, Benno Krojer, and Daniel Fried. 2023. Pragmatic inference with a CLIP listener for contrastive captioning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1904–1917, Toronto, Canada. Association for Computational Linguistics.

Tomáš Paus, David W Perry, Robert J Zatorre, Keith J Worsley, and Alan C Evans. 1996. Modulation of cerebral blood flow in the human auditory cortex during speech: Role of motor-to-sensory discharges. *European Journal of Neuroscience*, 8(11):2236–2246.

Martin J Pickering and Chiara Gambi. 2018. Predicting while comprehending language: A theory and review. *Psychological bulletin*, 144(10):1002.

Martin J Pickering and Simon Garrod. 2013. An integrated theory of language production and comprehension. *Behavioral and brain sciences*, 36(4):329–347.

Krishna Pillutla, Swabha Swayamdipta, Rowan Zellers, John Thickstun, Sean Welleck, Yejin Choi, and Zaid Harchaoui. 2021. Mauve: Measuring the gap between neural text and human text using divergence frontiers. *Advances in Neural Information Processing Systems*, 34:4816–4828.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR.

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

Seymour Rosenberg and Bertram D Cohen. 1964. Speakers' and listeners' processes in a word-communication task. *Science*, 145(3637):1201–1203.

Michael F Schober and Herbert H Clark. 1989. Understanding by addressees and overhearers. *Cognitive psychology*, 21(2):211–232.

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

Alane Suhr and Yoav Artzi. 2024. Continual learning for instruction following from realtime feedback. *Advances in Neural Information Processing Systems*, 36.

Shengbang Tong, Erik Jones, and Jacob Steinhardt. 2024a. Mass-producing failures of multimodal systems with language models. *Advances in Neural Information Processing Systems*, 36.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. 2024b. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9568–9578.

Ramakrishna Vedantam, Samy Bengio, Kevin Murphy, Devi Parikh, and Gal Chechik. 2017. Context-aware captions from context-agnostic supervision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 251–260.

Julia White, Jesse Mu, and Noah D. Goodman. 2020. Learning to refer informatively by amortizing pragmatic reasoning. *Preprint*, arXiv:2006.00418.

Deanna Wilkes-Gibbs and Herbert H Clark. 1992. Coordinating beliefs in conversation. *Journal of memory and language*, 31(2):183–194.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256.

Arianna Yuan, Will Monroe, Yu Bai, and Nate Kushman. 2018. Understanding the rational speech act model. In *CogSci*.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.

# A Training and Inference Details

## A.1 Training Hyperparameters

We use the instruction-tuned IDEFICS2-8B model (Laurençon et al., 2024) for all our experiments, and optimize with AdamW (Loshchilov and Hutter, 2019) with a learning rate of 0.0001 and a weight decay of 0.1. Each gradient step is computed over independently sampled minibatches of size 32 for comprehension and generation tasks.

We use LoRA for finetuning (Hu et al., 2022), where $r = 16$ and $\alpha = 8$. We apply adapters to all feedforward layers in the vision encoder, the modality projection and the perceiver-resampler block, but only to the key, query and value projections of the text decoder. We found applying further adapters for the text decoder to exacerbate overfitting. We load and train models with BF16 precision to reduce memory and compute costs.

We observe the IPS term for negatively rewarded examples (Section 3.2) to infrequently attain high values in early epochs during pilot experiments. To increase training stability, we clip the IPS term at 5. During our main experiment, clipping is activated for 2-3% of negatively rewarded generation examples in the first epoch, with the proportion declining afterwards.

## A.2 Stopping Criterion

Each model is trained for a maximum of 15 epochs. An epoch is a complete pass over data for the comprehension task. We use patience stopping, ending training when model validation accuracy for the comprehension task does not improve for five epochs. For models with joint inference, we compute validation accuracy with the joint listener model $P_l^j(t|\mathcal{I}, u; \theta)$, while for models without joint inference, we compute it with the base listener $P_l(t|\mathcal{I}, u; \theta)$. We exclusively use comprehension accuracy. Pilot experiments showed it correlates well with deployment performance.

## A.3 Hyperparameter Search

Hyperparameter search is done on the seed initialization data, using comprehension accuracy on the validation set as the metric. We vary learning rates $\{1e-5, 5e-5, 1e-4, 2e-4\}$, weight decay $\{0, 1e-3, 1e-1\}$, LoRA $\alpha$ $\{8, 32\}$ and adapter placements (only key-query-value projections; all forward projections; and all forward projections except for the language decoder, which used key-query-value projections) and prompt designs. The selection of LoRA adapters is the most important hyperparameter, showing a strong impact on overfitting at the data scales we work in.

The $\lambda_l$ hyperparameter for the joint comprehension distribution is tuned on the seed data with the hyperparameters described on Appendix A.1. We save model checkpoints for each epoch of training and inspect comprehension accuracy values on the validation set for different settings of $\lambda_l$. We find $\lambda_l = 0.5$ consistently perform well.

We choose $\lambda_s$ by training models with the joint inference strategy with $\lambda_l = 0.5$ and using the hyperparameters and stopping criterion from Appendix A.1 and Appendix A.2. We sample utterances on the validation set and inspect the reranking behavior of the joint generation distribution $P_s^j(u|\mathcal{I}, t; \theta)$ with different $\lambda_s$ values. We observe that the utterance the joint distribution $P_s^j(u|\mathcal{I}, t; \theta)$ ranked as the best was often equivalent to the utterance the base generation distribution $P_s(u|\mathcal{I}, t; \theta)$ ranked as the most likely. This skew towards the base generation distribution is additionally exacerbated with longer training times.

To determine $\lambda_s$ in light of this, we probe how accurately the joint generation distribution could

rank utterances on the validation set. Specifically, for each context-target pair on the validation set, we measure whether the distribution $P_s^j(u|\mathcal{I}, t; \theta)$ assigned higher probabilities to the ground-truth utterance for that pair than distractor utterances collected for other target images in that context. We vary $\lambda_s$ in $[0, 1]$ with increments of $0.01$ and find that $\lambda_s = 0$ achieved the best accuracy at selecting the ground-truth utterance.

### A.4 Prompt Design

We use the same model (i.e., same architectures and same parameters) for comprehension and generation and designate which task the model should perform through prompting. Figure 6 and Figure 7 show the prompts for comprehension and generation.

### A.5 Generation Sampling Details

We sample utterances autoregressively using a temperature of $\tau = 0.7$. We sample $k = 10$ utterances to generate with the joint inference procedure. To isolate the influence of reranking with the comprehension model, we also sample $k = 10$ utterances when not performing joint inference and return the utterance with the highest probability.

### A.6 Computational Resources

Each model is trained with a single GPU, RTX A6000 or NVIDIA A100. Hyperparameter tuning experiments took 100-200 GPU hours total, while training for the main continual learning experiment took approximately 225 GPU hours. For deployment, on the other hand, Models are deployed using RTX A6000 and V100 GPUs, with Ray for inference parallelization (Moritz et al., 2018).

## B Context Construction

Each reference game round involves a context of size $N = 10$ comprising 3 blocks (two of size 3 and one of size 4) of visually similar tangrams. We use a CLIP model (Radford et al., 2021) fine-tuned by Ji et al. (2022) on annotations from the KILOGRAM dataset to construct these sub-blocks. The blocks increase the difficulty of the context, because elements within each block have high visual similarity, making both comprehension and generation more challenging.

Each similarity block is constructed by randomly sampling a tangram, and sampling the rest of the block members from all other tangrams. The sampling is done using a distribution of normalized similarity scores between the first sampled tangrams and all other tangrams. The similarities are computed using CLIP.

## C Experiment Details

### C.1 Set of Spatial Reasoning Words

We curate the set of words relating to spatial reasoning by parsing the set of all human and model-generated utterances using spaCy with the `en_core_web_sm` pipeline (Honnibal et al., 2020). We collect the set of all words marked with an ADP (adposition) part-of-speech tag, which predominantly contained terms for spatial reasoning in our task, and manually filtered out the words such as "like" that are irrelevant to spatial reasoning. We then added words relating to notions of "left" and "right," which were not captured under the ADP tag.

The full set of words we used was: *'from', 'towards', 'thru', 'to', 'through', 'until', 'next', 'above', 'along', 'about', 'out', 'inside', 'behind', 'outside', 'forward', 'back', 'around', 'beneath', 'atop', 'up', 'apart', 'near', 'at', 'below', 'into', 'onto', 'toward', 'past', 'upwards', 'before', 'within', 'against', 'between', 'beside', 'on', 'after', 'by', 'over', 'across', 'down', 'opposite', 'underneath', 'in', 'under', 'left', 'leftward', 'leftwards', 'right', 'rightward', 'rightwards'*.

### C.2 Shape Naming Divergence Metric

We analyze pragmatic reasoning using the Shape Naming Divergence (SND) metric (Ji et al., 2022), which measures how much the naming of individual tangrams varies across different annotations. We repurpose it to probe pragmatic reasoning by measuring how much a model's description of a given tangram varies across different contexts. Instead of descriptions from different annotators, we compute SND over descriptions of that tangram in different contexts. This gives insight into the impact of the context (i.e., via pragmatic reasoning) on the description of the individual tangram.

## D Additional Performance Analyses

### D.1 Estimating Performance on Future Rounds

The decisions of experiment length (i.e., in the number of rounds) requires to balance costs and research utility. Our main experiment included four rounds of deployment and learning, which was sufficient to answer our research questions given the

**Comprehension Prompt**:
[User] You will be presented with a sequence of 10 images and a caption describing exactly one of them. Your task is to guess which image the caption describes. Image 0: <img0>, Image 1: <img1>, Image 2: <img2>, Image 3: <img3>, Image 4: <img4>, Image 5: <img5>, Image 6: <img6>, Image 7: <img7>, Image 8: <img8>, Image 9: <img9>. Caption: <speaker caption>. Does this caption describe Image 0, 1, 2, 3, 4, 5, 6, 7, 8 or 9?

[Assistant] The caption describes Image <target image index>

Figure 6: IDEFICS2 comprehension prompt. The target image index is not provided during inference time.

**Generation Prompt**:
[User] You will be presented with a sequence of 10 images and be assigned a target image. Your task is to produce a caption for your target image such that anyone could guess the image from your description. Image 0: <img0>, Image 1: <img1>, Image 2: <img2>, Image 3: <img3>, Image 4: <img4>, Image 5: <img5>, Image 6: <img6>, Image 7: <img7>, Image 8: <img8>, Image 9: <img9>. Your target is Image <image index>. Produce your caption now.

[Assistant] <caption>

Figure 7: IDEFICS2 generation prompt. The caption is not provided during inference time.

dramatic differences between the systems. Our data does allow us to estimate performance trends for one more round, without collecting additional data. We train models for a fifth round given all the interaction data collected in prior rounds, including the last round of deployment, which provided the final performance numbers. We compute offline estimate of comprehension performance using human-model interactions collected on the fourth round by the control system (i.e., the initial FULL model), which come from the same distribution of human utterances and are unseen by models in training.

This estimate indicates the trends we observe are robust, and continue for at least one more round beyond our experiment. Comprehension performance continues to improve for all models (FULL: 72.79 $\rightarrow$ 76.79%; NO-JI: 66.86 $\rightarrow$ 68.25%; NO-DS: 64.73 $\rightarrow$ 65.93%; BASELINE: 58.04 $\rightarrow$ 64.25%). FULL still outperforms all other systems by a large margin. Importantly, the performance of FULL on the second round remains larger than the improved performance of BASELINE (65.24% > 64.25%), validating our observation that coupling boosts data efficiency. This indicates that the positive impacts the coupling of comprehension and generation has on performance trends are likely to persist.

### D.2 Impact of Data Sharing on Training Set Size

Figure 8 shows how the number of datapoints models train on for comprehension and generation tasks change over time. Coupling with data sharing leads to a strong data augmentation effect for FULL and NO-JI, with the number of datapoints shared from the opposing role increasing as the model performance increases.

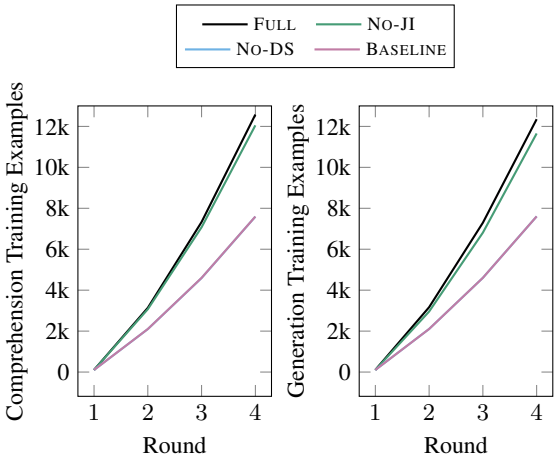


Figure 8: Number of training examples for comprehension and generation tasks across four rounds of deployment. The plots account for datapoints converted from the opposing role when data sharing is applied.

## E Crowdsourcing

### E.1 Worker Recruitment

We recruit workers with a minimum HIT (Human Intelligence Task) approval rate of 98% and at least 1,000 approved HITs. We restrict the pool to workers from English-majority locales (United States, Canada, Great Britain, Ireland, Australia, and New Zealand). Workers complete a video tutorial and a qualification quiz to qualify for our tasks.[11] The quiz also includes accepting a consent form. The

[11]The quiz may be found in our codebase. The video tutorial is accessible at https://lil-lab.github.io/tangrams-refgame-dev/.

consent form details how identifiable information of workers (i.e.,AMT worker IDs) is encrypted, how the collected data would be published, and benefits and risks from participating in the study. We recruit a total of 84 workers. This study was qualified as exempt by Cornell University's Institutional Review Board.

Even with the qualification process, workers that produced low-effort responses or colluded with others entered the worker pool. We further estimate the the effectiveness of workers via human-human games. We collected a set of 113 pilot games between humans, where at the end of each HIT, players rated their satisfaction with their partner on a Likert scale from 1–6. We removed workers with an average less than 4 from the pool and manually reviewed the games of the remaining workers. With this process, we restricted the pool of workers to a set of 50 experts, 41 of whom joined our final experiments. We collected our initialization and validation data, and performed our continual learning experiment with this set of experts.

### E.2 Payment Details

The HIT base pay is $0.60USD. For each round of reference games played within a HIT, workers receive a bonus of $0.125USD upon success or $0.05USD upon failure. The estimated hourly pay was $18.31 USD for games between humans, and $20.55 USD for games between humans and models at the final round. We set the base pay and bonuses through pilot studies among researchers and tuned the values based on estimates of hourly pay during pilot studies.

### E.3 Game Interface

The reference game interface is built using the Empirica framework (Almaatouq et al., 2021). It includes a chatbox at the left hand of the screen and the context tangrams at the center. When in the speaker role, the target is indicated to the speaker with a black square. The speaker has 45 seconds to type and send an utterance through the chatbox. After the speaker sends a message, the listener is given 15 additional seconds to make a selection. Each round lasts at most 60 seconds. The listener makes a selection by clicking on a tangram image. If successful, the target flashes green for both players. Upon failure, the target tangram flashes red for the speaker and the chosen tangram flashes red for the listener. Workers in the speaker role are not revealed their partners' choice and workers in the listener role are not revealed the target. We do this to mitigate worker adaptation to models and convention formation throughout a HIT. If neither player makes a decision within the given timeframe, the round is considered unsuccessful. The HIT terminates if an individual worker does not take an action for two consecutive rounds. Figure 9 shows the HIT introduction, listener role, and speaker role.

### E.4 Deployment Details

In each deployment, we give each worker access to an equal number of HITs to uniformly sample from the worker pool. Within a given HIT, a worker plays 40 rounds of reference games, either against a human or model partner. If playing against a model, the worker plays against each system variant an equal number of times and in a random order. During the final round of deployment, we additionally evaluate the initial FULL system, and therefore increase the number of rounds per HIT to 50.

Throughout the execution of a HIT, players alternate between roles every 3–4 rounds. In each group of 3–4 rounds, the underlying context is kept fixed, with the targets changing each round. This balances the cognitive load of observing a completely new context while preventing workers from being able to guess targets based on what has not been mentioned yet. If a worker is playing against a model, the system they are playing against is kept fixed within this group of 3–4 rounds. Workers are not revealed whether they are playing against a human or a model.

Each HIT additionally includes an attention check round at a random position. The attention checks are randomly sampled from a set of 100 manually annotated context-target pairs. To ensure simplicity, we sample the targets from the bottom 15th percentile of tangrams in terms of the SND metric (indicating high annotator agreement for tangram naming within the KILOGRAM dataset) and restrict the remaining tangrams in the context to those with a CLIP cosine similarity less than 0. The rest of the attention check construction follows the process outlined in Appendix B.

In practice, we did not disqualify any workers. Our main continual learning experiment spanned from May 3rd to May 21st.

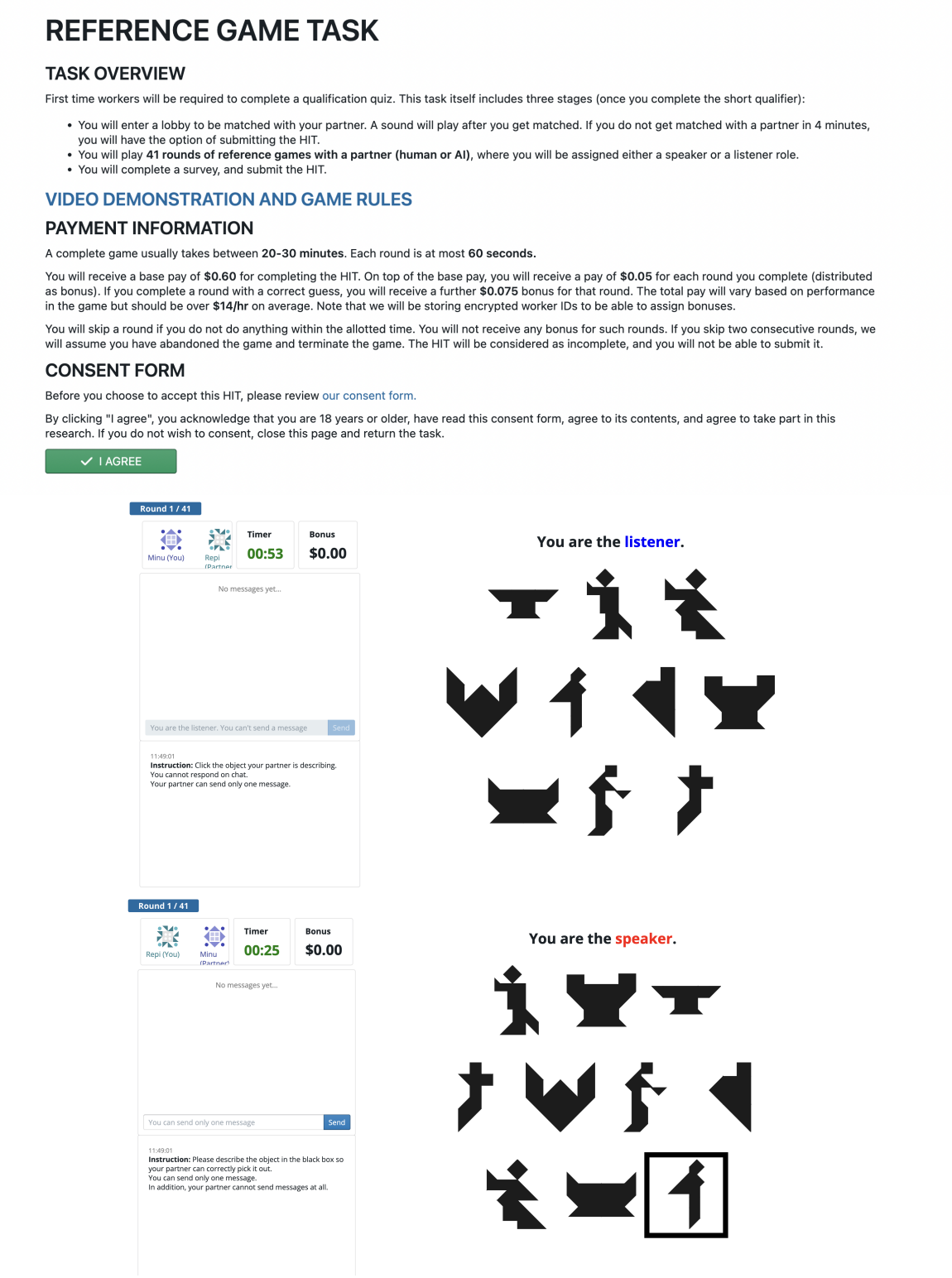


Figure 9: Top: the introduction screen shown upon accepting a HIT. Center: worker view in the listener role. Bottom: worker view in the speaker role.

# F Data Details

## F.1 Interactions Per Round

We collect 2,000 interactions for each role for each system in the first round, and increase the number by 500 each round (Section 5). For each role and each system, we collect 2,000 interactions on round 1, 2,500 on round 2, 3,000 on round 3, and 3,500 on round 4.

### F.2 Data Release

We release all of the data collected during our experiments alongside the code used to conduct them. This includes the seed training and validation sets of 104 and 280 successful human-human reference games as well as all of the interactions collected during continual learning, comprising 10,811 rounds of human-human reference games, and 43,442 and 43,492 rounds of human-model reference games where the model is in the listener or speaker roles. We do not include rounds where the human partner idled.

During data collection, all worker IDs were encrypted with MD5 hashes. Worker information is further anonymized during the release by mapping each ID hash to a numeric index.

## G Licenses of Scientific Artifacts Used

Our chosen model architecture, IDEFICS2-8B (Laurençon et al., 2024), and the Ray library have open licenses (Apache 2.0); the repository for MAUVE (Pillutla et al., 2021) has a GNU General Public License; and spaCy (Honnibal et al., 2020) has an MIT license.