# Generating Language Corrections for Teaching Physical Control Tasks

**Megha Srivastava** [1] **Noah Goodman** [1 2] **Dorsa Sadigh** [1]

## Abstract

AI assistance continues to help advance applications in education, from language learning to intelligent tutoring systems, yet current methods for providing students feedback are still quite limited. Most automatic feedback systems either provide binary correctness feedback, which may not help a student understand *how* to improve, or require hand-coding feedback templates, which may not generalize to new domains. This can be particularly challenging for physical control tasks, where the rich diversity in student behavior and specialized domains make it challenging to leverage general-purpose assistive tools for providing feedback. We design and build **CORGI**, a model trained to generate language corrections for physical control tasks, such as learning to ride a bike. **CORGI** takes in as input a pair of student and expert trajectories, and then generates natural language corrections to help the student improve. We collect and train **CORGI** over data from three diverse physical control tasks (drawing, steering, and joint movement). Through both automatic and human evaluations, we show that **CORGI** can (i) generate valid feedback for novel student trajectories, (ii) outperform baselines on domains with novel control dynamics, and (iii) improve student learning in an interactive drawing task.

## 1. Introduction

In our daily lives, we need to learn a variety of physical control tasks (e.g. driving a car or athletic sports) that benefit from receiving feedback of different modalities, such as visual demonstrations or haptic guidance. One of the most general forms of corrective feedback, however, is natural language – a person learning how to ride a bike can easily understand what *"make a sharper left turn"* means, even if they are unfamiliar with the specific control dynamics of the task. While recent works have focused on learning control policies that incorporate natural language feedback from users (Broad et al., 2017; Cui et al., 2023; Sharma et al., 2022), few have considered the reverse direction of automatically generating language corrections to provide to human users. Such corrections can be useful for enhancing human-AI interaction in decision making contexts (Lai & Tan, 2019), improving interactive data collection (Gandhi et al., 2022; Gopalan et al., 2022), and more generally teaching humans how to perform physical control tasks such as for rehabilitation, flying an aircraft, or operating surgical robots. (Hayws et al., 2009; Maciejasz et al., 2014; Srivastava et al., 2022; Yu et al., 2022; Schrum et al., 2022).

How do humans typically provide natural language feedback? Consider a parent who is teaching their child how to ride a bike. One form of corrective feedback they may provide are general, vague utterances (e.g. *"that was okay, try again"*) that provide positive or negative reinforcement, but may not be very informative on *how* to improve. On the other extreme, the parent may provide precise feedback (e.g. *"wider grip on the handle-bars"*) that clearly conveys how the child should adjust their behavior, but requires access to domain-specific information such as referring to handle bars, which is only applicable to the setting of teaching how to ride a bike. This results in a trade-off between helpfulness, or the ability to provide sufficient information to help a student improve, of corrections and their generality, or ability to be understood and conveyed across different settings.

In fact, existing works on automatic feedback generation in domains such as programming and language learning reflect this trade-off (Settles et al., 2020; Liu et al., 2022). Some systems provide simple binary feedback (e.g. whether a program ran successfully), which may not be very helpful to the student, while others require hand-coded, templates (e.g. grammar checking) that lack generality. Due to the rich diversity of physical control tasks and variation in ways a student might under-perform, we seek to strike a balance by learning to generate helpful comparative corrections (e.g. *"brake sooner"*) that can also generalize to novel trajectories within the same control space. To achieve this, we choose to leverage the expressive capabilities of language models (LMs), driven by the key insight that LMs may encode phys-

[*]Equal contribution [1]Department of Computer Science, Stanford University [2]Department of Psychology, Stanford University. Correspondence to: Megha Srivastava <megha@cs.stanford.edu>.

*Proceedings of the $40^{th}$ International Conference on Machine Learning*, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

ical conceptual spaces that are isomorphic across the variety of environments, states, and action spaces that exist across different physical control tasks (Patel & Pavlick, 2022).

Concretely, we design and build **CORGI**, a model trained for natural language **cor**rections **g**eneration for the **i**nstruction of three physical control tasks: DRAWING, STEERING a vehicle, and body MOVEMENT. These three tasks exhibit different control spaces such as 2D x-y surface position, steering and acceleration, and skeleton joint motion, which require **CORGI** to develop a general understanding of physical concepts. At test time, **CORGI** takes in as input a pair of student and expert trajectories, and generates a correction in natural language to help the student better match the expert's performance. Specifically, **CORGI** consists of a trainable trajectory encoder that learns to map student and expert trajectories to prompts that can be used as inputs to a frozen LM to generate feedback with, thus keeping the more general representations of language encoded by the LM fixed. Through both automatic and human evaluations, we show that **CORGI** can (i) generate valid feedback for novel student trajectories, (ii) outperform baselines on domains with novel control dynamics, and (iii) improve student learning in an interactive drawing task. Our full contributions include:

1. A dataset of 2k crowdsourced corrections collected across (student, expert) trajectories from a diverse set of control tasks (drawing, steering, and joint motion).
2. **CORGI**, our model trained to generate corrective feedback in natural language for these three tasks.
3. A comprehensive evaluation of the ability of **CORGI** to generalize to novel student trajectories and domains that share the same control space.
4. Two human subject user studies assessing both preference and the helpfulness of generated feedback in helping users improve drawing.

## 2. Related Works

While recent works have explored generating *comparative* descriptions, such as language descriptions of distribution shifts (Zhong et al., 2022) and relative image captions (Mirchandani et al., 2022), we are the first to explore this for physical control tasks, as well as with an educational focus.

**Language in Multimodal Tasks** Several works have leveraged advances in LMs and multimodal models to improve human interaction across physical control tasks. For example, Google's SayCan leverages LMs to break down language instructions into executable skills, providing users flexibility in receiving robotic assistance for complex, long-horizon tasks (Ahn et al., 2022). Others have explored using language to adjust robot plans with constraints or specify subgoals (Sharma et al., 2022; Cui et al., 2023). Finally, (Tevet et al., 2022) recently introduced MotionCLIP, a transformer-based auto-encoder that enables exciting text-to-motion capabilities, such as adjusting human motion sequences for novel styles (e.g. *"run away hysterically"*).

Another multimodal task closely related to ours is image (or video) captioning, where large-scale multimodal models have achieved state-of-the-art performance on classic benchmarks such as MSCOCO (Alayrac & et. al., 2022; Lin et al., 2014). Furthermore, Tsimpoukelli et al. (2021) achieve strong performance on captioning tasks by only training a visual encoder to output a prompt for a frozen LM, motivating our approach for **CORGI**.

**Language in Education** A few works have studied the role of language descriptions and feedback in educational settings. Chopra et al. (2019) show that language can reduce time in communicating concepts to a student, Sumers et al. (2020) find in a cooperative teaching game that language helps communicate more nuanced concepts than other feedback forms like demonstrations, and Ruan et al. (2019) demonstrate that interactive dialogue-based agents can improve student learning. However, these works largely focus on understanding the role of language in pedagogical settings, not automatically generating language feedback.

**Language in Physical Interaction Datasets** Large-scale datasets of language paired with physical interactions have enabled further understanding of physical reasoning, as well as inspired progress on novel interactive control tasks. For example, Ji (2022) built a rich-annotated dataset of tangram puzzles to study the abstract visual reasoning capabilities of multi-modal models, Wong et al. (2022) show how to leverage annotations in the CLEVR dataset (Johnson et al., 2017) to improve generalization on spatial relationship tasks and Lynch & Sermanet (2021) show that "play" data annotations enable strong zero-shot language conditioning for robotic tasks. To the best of our knowledge, we are the first to collect corrections over pairwise trajectories, providing insight into how people reason about physical comparisons.

## 3. Generating Corrective Feedback

We now formalize generating corrective feedback in an educational setting, where the goal is to generate corrections from the set of possible natural language utterances $u \in \mathcal{U}$ that are *comparative* with respect to some expert behavior. Consider a target physical control task $g$ (e.g. riding a bike), a student $\mathcal{S}$ (e.g a child learning to ride a bike), and an expert $\mathcal{E}$ (e.g. their parent who can already perform this task). We can treat $g$ as a standard Markov decision process (MDP) $< S, A, f, R, T >$ with finite horizon $T$, reward function $R : S \times A \rightarrow \mathbb{R}$ over state $S$ and action $A$ spaces, and

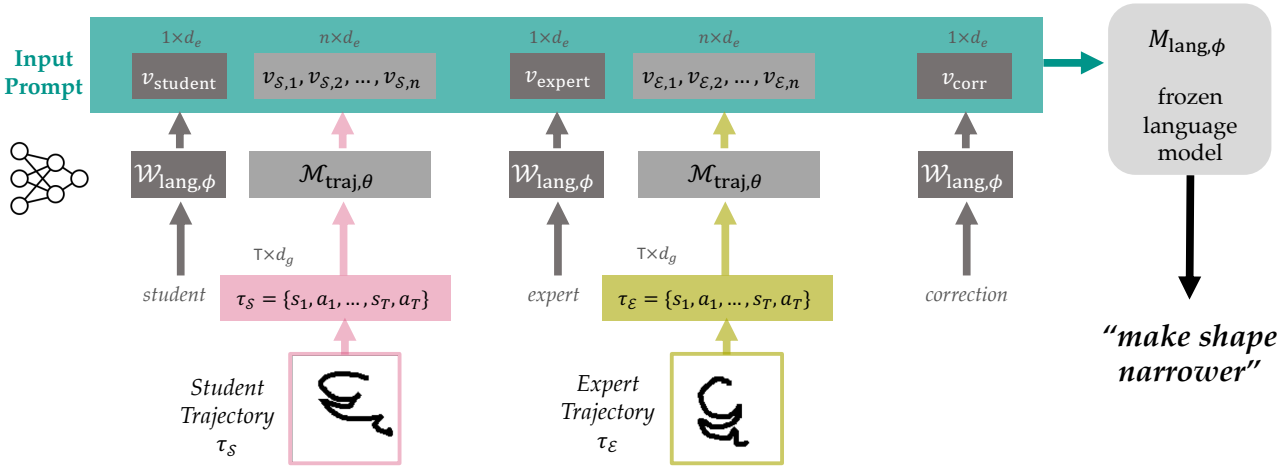


*Figure 1.* Overview of **CORGI** at test time. Trajectories $\tau_{\mathcal{S}}$, $\tau_{\mathcal{E}}$, from a student and an expert respectively, are mapped by a learned trajectory encoder $M_{\text{traj},\theta}$ to vectors of the same dimension as the output of the frozen language model $M_{\text{lang},\phi}$'s embedding layer ($W_{\text{lang},\phi}$). The resulting output vectors are stitched together with the embeddings corresponding to vocabulary words "student", "expert", and "correction" in order to create the input prompt sent to the $M_{\text{traj},\theta}$, from which we then generate a correction.

a deterministic transition function $f : S \times A \rightarrow S$ that maps a particular state and action pair $s_t, a_t$ at time step $t$ to a new state $s_{t+1}$. We can then define a trajectory $\tau$ as a sequence of state and action pairs $\{s_1, a_1, \ldots, s_T, a_T\}$, and can collect trajectories from both the student ($\tau_{\mathcal{S}}$) and the expert ($\tau_{\mathcal{E}}$). Under this setting, we now formalize the goal of generating corrective feedback $u$ for the student $\mathcal{S}$.

### 3.1. Problem Statement

Effective feedback should reduce discrepancies between a student learner's current understanding and performance of a task and that of an expert teacher (Hattie & Timperley, 2007). Therefore, good corrections should not only accurately identify such discrepancies, but also be sufficiently *helpful* for the student to improve. We thus assess a correction $u$ by measuring the degree it reduces the gap between the student $\mathcal{S}$'s and expert $\mathcal{E}$'s performance on task $g$.

Concretely, let $\pi^k_{\mathcal{S},g}$ and $\pi_{\mathcal{E},g}$ represent the student and fixed expert policies for task $g$ at time $k$ from which we can collect trajectory rollouts $\tau^{g,k}_{\mathcal{S}}$ and $\tau^g_{\mathcal{E}}$. Furthermore, let $\mathcal{L}$ be a task-dependent loss function that measures the discrepancy between two trajectories. A corrective feedback utterance $u_k$ provided at timestep $k$ may result in the student updating their policy from $\pi^k_{\mathcal{S},g}$ to $\pi^{k+1}_{\mathcal{S},g}$, and so the optimal corrective feedback would be a $u_k$ that minimizes the expression:

$$\min_{u_k} \mathcal{L}(\tau^{g,k+1}_{\mathcal{S}}(u_k), \tau^g_{\mathcal{E}}) - \mathcal{L}(\tau^{g,k}_{\mathcal{S}}, \tau^g_{\mathcal{E}}) \tag{1}$$

In other words, our goal is to generate language corrections $u$ that result in the largest decrease in discrepancy between the student and the expert. In practice, however, optimizing directly for the above expression is intractable due to the lack of strong cognitive models of human learning, i.e., we do not have an accurate model of how $u_k$ leads to changes in the student trajectory $\tau^{g,k+1}_{\mathcal{S}}$. Therefore, instead of optimizing for the objective in Eq. (1), we consider whether it is possible to build a strong generative model in a supervised manner from annotated samples of corrective feedback $(\tau^g_{\mathcal{S}}, \tau^g_{\mathcal{E}}, u)$. In order to best capture the expressiveness of annotations provided in natural language, we propose leveraging the rich encoding of language present in modern day LMs by casting the problem of generating corrective feedback for student $\mathcal{S}$ in reference to $\mathcal{E}$ as a *controllable text generation* problem. Concretely, our goal is to identify a method that, given tuples of $(\tau^g_{\mathcal{S}}, \tau^g_{\mathcal{E}}, u)$, allows us to effectively control (via prompting) a large pretrained LM to generate corrections $u$ at test time when we only have access to novel student and expert trajectories $(\tau^g_{\mathcal{S}}, \tau^g_{\mathcal{E}})$.

### 3.2. Trajectory Encoding

To use trajectory samples $(\tau^g_{\mathcal{S}}, \tau^g_{\mathcal{E}})$ to construct an input prompt that can help steer an LM to generate good corrections $u$, we first need the ability to represent trajectories of a physical task as a sequence of text tokens. Recall that a trajectory $\tau$ is a sequence of state and action pairs $\{s_1, a_1, \ldots, s_T, a_T\}$ which, when concatenated can be represented as a set of $T$ vectors of numerical values with dimension $d_g := [S] + [A]$. Meanwhile, a typical LM ($\mathcal{M}_{\text{lang},\phi}$) consists of a word embedding layer ($\mathcal{W}_{\text{lang},\phi}$) that maps text tokens from a fixed vocabulary to embeddings of a given dimension $d_e$. We therefore learn a trajectory encoder model $\mathcal{M}_{\text{traj},\theta}$ that can map any $(T \times d_g)$-dimension

trajectory $\tau^g$ to a set of $n$ vectors of dimension $d_e$, where $n$ is a hyperparameter. We can then represent $\tau^g_{\mathcal{E}}$ and $\tau^g_{\mathcal{S}}$ as a sequence of "token embeddings" $v_{\mathcal{S},1}...v_{\mathcal{S},n}$, $v_{\mathcal{E},1}...v_{\mathcal{E},n}$ that, as shown in Figure 1, form the input prompt to the LM which we will use to conditionally generate correction $u$.

### 3.3. Controllable Text Generation

**CORGI** consists of a trainable encoder $\mathcal{M}_{\text{traj},\theta}$ that learns to represent any arbitrary trajectory $\tau$ as a sequence of continuous embeddings such that, when embeddings corresponding to both the student and expert trajectories are included as part of a prompt, the underlying *frozen*, pre-trained LM ($\mathcal{M}_{\text{lang},\phi}$) will generate appropriate corrections. We choose to keep the LM frozen in order to aid the adaptability of **CORGI** to new kinds of student behavior and domains where there may be changes in language not captured by our data.

We learn the same trajectory encoder ($\mathcal{M}_{\text{traj},\theta}$), consisting of a 3-layer feed-forward neural network that outputs $n$ vectors with the same dimension as the target LM (e.g. 768 for GPT-2), for both student $\mathcal{S}$ and expert $\mathcal{E}$ trajectories. We train our model over tuples of corrections paired with student and expert trajectories $(\tau_{\mathcal{S}}, \tau_{\mathcal{E}}, u)_i$ by constructing input prompt sequences using $\mathcal{M}_{\text{traj},\theta}$ as shown in Figure 1. During training, we calculate the language modeling loss, where the loss of single sample $q_i$ is:

$$\mathcal{L}_\phi(q_i) = -\sum_{t=1}^{|q_i|} \log \mathcal{M}_{\text{lang},\phi}(q_{i_t}|q_{i_{<t}})$$

However, we only use $\mathcal{L}_\phi(q_i)$ to update weights $\theta$ of the trajectory encoder $\mathcal{M}_{\text{traj},\theta}$, keeping the weights of $\mathcal{M}_{\text{lang},\phi}$ frozen. At test time, we use the same format (omitting $u$ which is unknown) to construct the input prompt provided to the frozen LM from which we generate corrections.

### 3.4. Annotating Corrections & Data Augmentation

In order to train **CORGI**, we need to collect data of corrections for paired trajectories. Because our goal is for **CORGI** to generalize well to novel trajectories and domains, we are primarily interested in shorter, general corrections that do not refer to specific aspects of the expert's trajectory or domain-specific objects. Concretely, we ask annotators to provide brief samples of corrective feedback $u^{(1)}, u^{(2)}, ..., u^{(m)}$ for a particular $\tau^g_{\mathcal{S}}, \tau^g_{\mathcal{E}}$ trajectory pair for task $g$ in free-form text, encouraging annotators to identify which of the potentially several different ways for the student to improve they believe is most optimal to describe. We can then use tuples $(\tau^g_{\mathcal{S}}, \tau^g_{\mathcal{E}}, u^{(i)})$ to construct input prompts to train **CORGI**. Further details on crowdsourcing results of for our annotation procedure are described in Section 4.2 .

However, we observe that when human annotators provide corrective feedback in natural language, there exists greater variance in the language style of the provided corrections than the particular discrepancies they refer to. In order to enable **CORGI** to better capture this rich style diversity efficiently, we leverage more powerful, "instruction tuned" language models (e.g. OpenAI's `text-davinci-003`) for data augmentation. As described in Algorithm 1, for each annotation $u^{(i)}$ in our original dataset, we construct an input prompt describing a teaching setting and directly asking for paraphrases of $u^{(i)}$, which, when sent as input to a large instruction-tuned LM results in an augmented set of utterances $\{u_1'^{(i)}, u_2'^{(i)}, u_3'^{(i)}\}$ which are used for training. The prompt and example paraphrases are shown below:

**Algorithm 1** Train **CORGI**

**Input:** dataset $\mathcal{D}$ of $(u, \tau^g_{\mathcal{S}}, \tau^g_{\mathcal{E}})$ tuples with size $|\mathcal{D}|$
**Input:** frozen LM $\mathcal{M}_{\text{lang},\phi}$ with token embedding layer $\mathcal{W}_{\text{lang},\phi}$ and instruction-tuned LM $\mathcal{M}'_{\text{lang},\psi}$
**Input:** number of epochs $n_e$ , learning rate $\lambda$
Initialize trajectory encoder $\mathcal{M}_{\text{traj},\theta}$
`// data augmentation`
Set dataset $\mathcal{D}' \leftarrow \mathcal{D}$
**for** sample $i = 1$ **to** $|\mathcal{D}'|$ **do**
  Set prompt $p_i \leftarrow$ *"You are a teacher providing"* + *"feedback to a student learning a control task."* + *"List 3 short paraphrases of the feedback"* + $u_i$
  Set paraphrases $u'_{i,1}, u'_{i,2}, u'_{i,3} \leftarrow \mathcal{M}'_{\text{lang},\psi}(p_i)$
  $\mathcal{D}'$.append$((u'_{i,1}, \tau^g_{\mathcal{S}_i}, \tau^g_{\mathcal{E}_i}))$
  $\mathcal{D}'$.append$((u'_{i,2}, \tau^g_{\mathcal{S}_i}, \tau^g_{\mathcal{E}_i}))$
  $\mathcal{D}'$.append$((u'_{i,3}, \tau^g_{\mathcal{S}_i}, \tau^g_{\mathcal{E}_i}))$
**end for**
`// training`
**for** epoch $m = 1$ **to** $n_e$ **do**
  Shuffle dataset $\mathcal{D}'$
  **for** sample $i = 1$ **to** $|\mathcal{D}'|$ **do**
    Set prompt $q_i \leftarrow \mathcal{W}_{\text{lang},\phi}(\textit{student}) + M_{traj,\theta}(\tau^g_{\mathcal{S}_i}) + \mathcal{W}_{\text{lang},\phi}(\textit{expert}) + M_{traj,\theta}(\tau^g_{\mathcal{E}_i}) + \mathcal{W}_{\text{lang},\phi}(\textit{correction:}) + \mathcal{W}_{\text{lang},\phi}(u_i)$
    Set loss $\mathcal{L}(u_i, \tau^g_{\mathcal{S}_i}, \tau^g_{\mathcal{E}_i}) \leftarrow \mathcal{L}_\phi(q_i)$ `LM loss`
    Update $\phi \leftarrow \phi + \lambda \nabla_\theta \mathcal{L}(u_i, \tau^g_{\mathcal{S}_i}, \tau^g_{\mathcal{E}_i})$
  **end for**
**end for**

**annotator correction:**
turner slightly later $(u)$
**input prompt:**
You are a teacher providing feedback to a student learning a control task. List 3 short paraphrases of the feedback *"turner slightly later"*
**text-davinci-003 output:**
1. Make your turn a bit later. $(u'_1)$
2. Delay your turn a bit $(u'_2)$
3. Wait a moment before turning $(u'_3)$

The above example shows that paraphrases returned from the `text-davinci-003` LM retain the particular discrepancy of the correction while modifying its style, lan-

guage, and correcting for typos and grammatical errors. As we will show next (Table 1), training **CORGI** over augmented data improves performance across all control tasks.

## 4. Experimental Results

We now present our three tasks and experimental results. Details of user studies (including IRB approval) and training of **CORGI**, which is built on the 124M parameter version of GPT-2 (Wolf et al., 2019b), are in the Appendix.

### 4.1. Environments & Datasets

We study three physical control tasks that span common primitives: drawing (x-y control), steering (acceleration and heading angle control), and human body movement (joint control). For each environment, we also create in-domain (ID) and an out-of-domain (OOD) splits that share the same control space, but require different dynamics.[1]

DRAWING: The student's goal is to learn how to draw characters from different alphabet scripts. We select 10 characters from 5 scripts (ID: Arabic, Burmese, & Japanese, OOD: Futurama & Bengali) from the Omniglot dataset (Lake et al., 2015). We select 1 trajectory per character as the expert trajectory and randomly sample 5 student trajectories, split between train/test sets. Each trajectory is a sequence of 2D actions along x-y coordinates.

STEERING: The student's goal is to learn how to park a vehicle in a target parking spot. We modify the Parking environment from Leurent (2018) by changing the steering sensitivity and min/max speed for 3 vehicle types (ID: Car & Plane, OOD: Bike). For each vehicle type, we design a hand-coded expert policy, and then collect 20 student trajectories including perturbations of the expert policy and half-trained RL agents (details in Appendix A.3). Trajectories are split between train/test sets, and consist of 2D actions controlling acceleration and heading angle and 6D states corresponding to vehicle position, velocity, and heading.

MOVEMENT: The student's goal is to learn how to perform a full-body movement activity. We select activities from the BABEL dataset (Punnakkal et al., 2021) of 3D human motion (ID: Walk, Jump, & Throw, OOD: Wave, Jumping Jacks). For each activity we select 1 trajectory as the Expert, and sample 15 student trajectories, which are then split between train/test sets. We represent trajectories with learned video-text representations from X-CLIP (Ma et al., 2022), treating the output as a trajectory sequence of 1D states.

Example student trajectories for each environment are shown in Figure 2. We pad trajectories to a fixed dimension of 10 and length of 600 as input to **CORGI**. Further details on expert trajectory selection, as well as the assumption of a single expert behavior, are in Appendix A.2.

[1]While we aimed to pick OOD splits that were semantically far (e.g. Futurama is a synthetic language), it is still possible there may be smaller "sub-skills" shared between ID-OOD splits.

### 4.2. Crowdsourcing Details

We recruit crowdworkers on Prolific[2] to annotate paired student/expert trajectories with corrections. We instruct crowdworkers to not refer to expert demonstrations in their annotations. Crowdsourced corrections demonstrate a variety of ways people express feedback, such as rich shape descriptions (e.g. *"go towards making an infinity shape rather than a venn diagram"*), encouragement (e.g. *"more vertical but good effort"* ), and action ordering (e.g. *"after second bend draw towards left not down"*. We collect **2,023** corrections, and provide further details in Appendix A.4.

### 4.3. Automatic Evaluation

Our first evaluation goal is to measure the degree **CORGI** assigns high likelihood to examples of good corrections, which can be useful for tasks such as automatically evaluating feedback provided by instructors. In Table 1, we report the average perplexity (i.e. the exponentiated loss) across ground truth coorrections for novel student trajectories unseen during training, and for both ID and OOD splits of each task. We compare results across the following ablations:

- **Permute Correction**: Instead of conditionally generating a correction, we draw a random corrections from the same distribution as the ground-truth corrections – if a task has low variance across the types of feedback needed (e.g. all students need to "improve posture" in MOVEMENT), we should observe no difference.
- **Permute Student**: We simulate the setting where **CORGI** provides corrective feedback for a different student trajectory. This measures the degree **CORGI** may have only fitted to the fixed expert trajectory – it should assign higher (worse) perplexity when the student trajectory is randomized, showing the ability to tailor corrective feedback to individual students. For fair comparison, we sample student trajectories from the eval set to maintain the same overall distribution.
- **CORGI w/o Pretraining** : We able the effect of pretraining by (i) using the same GPT-2 architecture, but without pre-trained weights, and (ii) using a 3-layer LSTM with pre-trained embedding layer.
- **CORGI w/o Data Augmentation**: We train **CORGI** on the original, smaller dataset consisting purely of human annotations, without any paraphrases from our automatic data augmentation procedure.

[2]https://www.prolific.co/

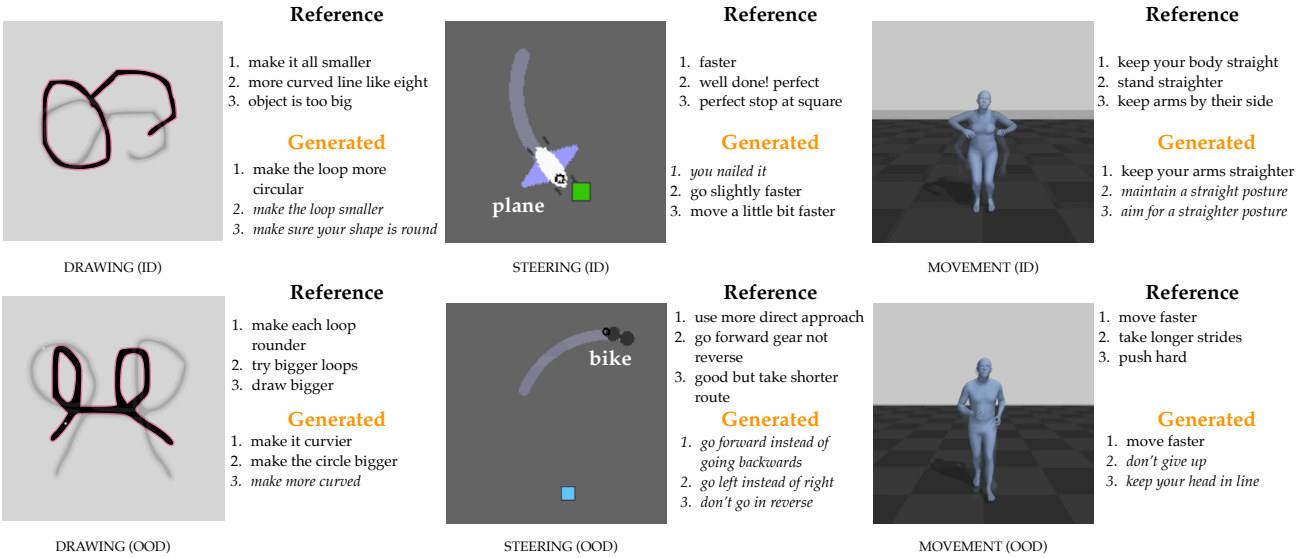


*Figure 2.* Example student trajectories, reference corrections from annotators, and corrections generated by **CORGI** for novel trajectories for all three control tasks. Generated corrections in *italics* are completely unseen during training, for any trajectory.

*Table 1.* Perplexity on held-out test sets (lower is better) across three control tasks. **CORGI** achieves lower perplexity in comparison to baselines across all tasks, and both pre-training and data augmentation components improve performance. Although there exists a gap between in-domain (ID) and out-of-domain (OOD) performance, **CORGI** still outperforms ablations even in OOD settings.

| Ablation | Drawing | | Steering | | Movement | |
|---|---|---|---|---|---|---|
| | ID | OOD | ID | OOD | ID | OOD |
| Permute Correction | 310 ± 38 | 249 ± 1.1 | 84 ± 18.5 | **194 ± 2.4** | 47 ± 2.3 | 123 ± 7.4 |
| Permute Student | 153 ± 5.6 | 256 ± 5.9 | 96 ± 8.9 | 218 ± 3.1 | 35 ± 0.28 | 111 ± 4.9 |
| **CORGI** | **145 ± 1.5** | **246 ± 2.5** | **51 ± 5.9** | **194 ± 2.3** | **33 ± 0.22** | **109 ± 3.1** |
| w/o Data Aug. | 162 ± 6.3 | 251 ± 2.9 | 54 ± 1.8 | 635 ± 24.3 | 36 ± 2.3 | 159 ± 6.7 |
| w/o Pretraining (GPT-2) | 959 ± 62 | 808 ± 72 | 302 ± 32 | 848 ± 88 | 376 ± 37 | 823 ± 53 |
| w/o Pretraining (LSTM) | 215 ± 1.2 | 584 ± 1.2 | 197 ± 1.4 | 271 ± 1.1 | 221 ± 1.3 | 252 ± 1.1 |

As Table 1 shows, **CORGI** outperforms both permutation ablations, suggesting that the model does take into account specific student trajectories, rather than just learning general task language. As expected, no pre-training decreases performance, due to the lack of strong language representations. Furthermore, data augmentation results in an improvement across all tasks for both ID and OOD settings. Although the gap between ID and OOD is high, we note that even in OOD settings **CORGI** generally outperforms ablations.

Thus, our second automatic evaluation focuses on the quality of generated samples from **CORGI**. Under a fixed set of decoding parameters (nucleus sampling (Holtzman et al., 2020), temperature = 0.5), we measure the average similarity between generated and ground-truth corrections across each $(\tau_{\mathcal{S}}, \tau_{\mathcal{E}})_i$ in our test set. However, as Figure 2 shows, annotations for a sample may have high variance due to identifying different discrepencies. We therefore use a re-weighted version of BERTScore that accounts for intrinsic variance between ground-truth captions, originally proposed for image captioning (Yi et al., 2020). In addition to the pre-training and data augmentation ablations, we compare the average similarity across generated samples from three alternative methods with **CORGI**:

- **Random:** We select a random human annotation from the same domain as the input trajectories, allowing us to the degree **CORGI**'s performance is due to just using vocabulary appropriate for the domain.
- **Nearest Neighbors:** For a given student trajectory in our test data, we use our trajectory encoder $\mathcal{M}_{\text{traj},\theta}$ to find the nearest neighbor student trajectory seen during training (using the mean squared error in encoder output). We then randomly sample from the set of ground-truth annotations provided for this student.

*Table 2.* Similarity scores on held-out test sets (higher is better) based on an improved BERTScore to account for ground truth variance from (Yi et al., 2020). Across all tasks, **CORGI** outperforms both randomly sampling ID feedback and a nearest neighbors baselines.

| METHOD | DRAWING | | STEERING | | MOVEMENT | |
|---|---|---|---|---|---|---|
| | ID | OOD | ID | OOD | ID | OOD |
| RANDOM | $0.20 \pm 0.03$ | $0.21 \pm 0.04$ | $0.19 \pm 0.04$ | $0.22 \pm 0.03$ | $0.23 \pm 0.06$ | $0.18 \pm 0.03$ |
| NEAREST NEIGHBORS | $0.28 \pm 0.03$ | $0.22 \pm 0.03$ | $0.28 \pm 0.05$ | $0.16 \pm 0.04$ | $0.31 \pm 0.05$ | $0.19 \pm 0.05$ |
| PERMUTE STUDENT | $0.22 \pm 0.03$ | $0.23 \pm 0.04$ | $0.14 \pm 0.03$ | $0.26 \pm 0.01$ | $0.14 \pm 0.03$ | $0.15 \pm 0.03$ |
| **CORGI** | $0.3 \pm 0.01$ | $\mathbf{0.34 \pm 0.03}$ | $\mathbf{0.32 \pm 0.08}$ | $\mathbf{0.31 \pm 0.02}$ | $\mathbf{0.39 \pm 0.03}$ | $\mathbf{0.24 \pm 0.03}$ |
| W/O PRETRAINING (GPT-2) | $0.11 \pm 0.02$ | $0.18 \pm 0.03$ | $0.10 \pm 0.03$ | $0.12 \pm 0.03$ | $0.11 \pm 0.03$ | $0.11 \pm 0.02$ |
| W/O PRETRAINING (LSTM) | $0.15 \pm 0.03$ | $0.17 \pm 0.03$ | $0.12 \pm 0.04$ | $0.13 \pm 0.03$ | $0.15 \pm 0.03$ | $0.18 \pm 0.02$ |
| W/O DATA AUG. | $\mathbf{0.32 \pm 0.04}$ | $0.26 \pm 0.04$ | $0.26 \pm 0.03$ | $0.27 \pm 0.02$ | $0.19 \pm 0.05$ | $0.23 \pm 0.02$ |

- **Permute Student:** We select a correction from the same domain and expert as the input trajectories, but a random student. Note this method is distinct from the Permute Student method in the previous section.

Table 2 shows that **CORGI** outperforms both methods across all tasks, for both ID and OOD settings. As expected, removing pre-training results in samples with lower similarity scores than **Random**, and we observe that without using a pre-trained LM, the model can only generate domain specific verbs (e.g. *"make"* or *"move"*). Interestingly, we observe that for this metric, there is less of a gap between ID and OOD – in fact, for DRAWING, generated samples from **CORGI** are *more* similar to ground truth annotations for OOD characters. As shown in Figure 2, for both ID and OOD we observe that **CORGI** indeed often generates corrections that are similar to the ground-truth annotations.

**Error Analysis**

In practice, however, neither automatic evaluation metric we report fully captures the complexities of evaluating corrections. For example, the types of sequences **CORGI** assigns high (worse) perplexity to include metaphorical utterances and noise (e.g. *"the shape at the top should be larger, marching the hook shape"*) and domain-specific language (e.g. *"go forward gear not reverse"*). Meanwhile, the improved BERTScore method from Yi et al. (2020) assigns a score of 0.0 to examples such as *(**reference**: well done, perfect!, **CORGI** : you nailed it!)*, where the expressed meanings are equivalent, but use very different language. This motivates the need for human evaluation, which we focus on next.

### 4.4. Human Preference Evaluation

We first choose to assess the degree human evaluators *prefer* **CORGI** over randomly chosen utterances from the same domain. Specifically, we measure preference as the rate at which human evaluators prefer the correction that is generated by **CORGI** when provided two other randomly selected corrections from the same domain. We then compare this rate with three other conditions that replace **CORGI**:

- **Random:** We calculate the rate at which human evaluators pick a correction randomly selected from the training data within the same domain. Since the other options are also randomly sampled, as the number of samples increase, this should converge to 33%.
- **Nearest Neighbors:** Already described in section 4.3, we randomly sample a ground-truth correction provided to the nearest neighbor student.
- **Ground Truth:** We calculate the rate at which human evaluators pick a corrections sampled from the set of ground-truth annotations for the target trajectory.

Users are shown a pair of student and expert trajectories (e.g. videos of human movement for MOVEMENT) and asked to pick one of the three corrections in response to the instruction *"Which feedback do you think is most helpful to provide to the student?"*. We collect preference data from 15 users per condition for each of our three tasks, randomizing the order in which each correction is provided. We recruit crowdworkers on Prolific, and provide further details in Appendix A.5. Due to cost, we limit ourselves to only novel in-domain (ID) trajectories for each of our control tasks.

Figure 3 shows that across all three control tasks, users were significantly more likely to prefer corrections from **CORGI** than our **Random** control. Furthermore, corrections generated with the **Nearest Neighbors** method are only comparable to those of **CORGI** for the MOVEMENT task, highlighting the ability of **CORGI** to generalize to student trajectories unseen during training. Surprisingly, in the STEERING task, we observe that **CORGI** significantly outperforms **Ground Truth**. One potential hypothesis is that preferences capture important aspects of corrections beyond accuracy, including clarity, constructiveness, and tone. Generated samples from **CORGI** are often concise and formal, while human corrections exhibit more variety. For example, the most common human annotation that evaluators did *not* select in the STEERING task was *"right hand down, route south"*, which may be less clear than the generated sample

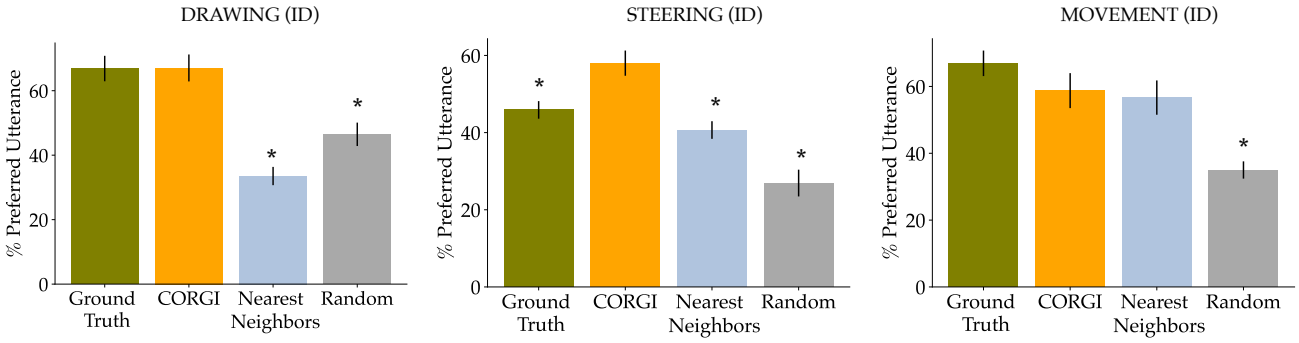


*Figure 3.* Across all three tasks, users are more likely to prefer feedback generated from **CORGI** over random corrections than feedback from a random control and nearest neighbors baseline. For STEERING, feedback from **CORGI** also outperforms ground truth corrections, which may be due to the high variance human annotations. Asterisk (*) marks statistically significant difference ($p < 0.05$) from **CORGI** .

for the same comparison (*"glide gracefully to the left"*). Finally, we provide pair-wise comparison results on feedback from **CORGI** when directly compared with **Ground Truth** and **Nearest Neighbors** feedback in Appendix A.5.

### 4.5. Learning from Feedback

Our final human evaluation directly measures the degree **CORGI** helps reduce the discrepancy between student $\mathcal{S}$ and expert $\mathcal{E}$ performance in the DRAWING task. We design a teaching interface, shown in Appendix A.6, where users are given three chances to draw a provided stimulus and match a hidden expert trajectory $\tau_{\mathcal{E}}$. The only information users receive are corrections corresponding to their trajectory $\tau_{\mathcal{S}}$, and a numerical score calculated with the mean squared error between $\tau_{\mathcal{S}}$ and expert trajectory $\tau_{\mathcal{E}}$. We then measure the change in student error between the first and third trial.

We assign 20 users to a control group where corrections are randomly sampled from data within the same domain, 20 users to a control group where no corrections are provided, and 20 users to the experiment group, who receive corrective feedback from **CORGI**. While users who received random feedback (**-0.17** $\pm$ **1.16**) and no feedback (**-0.20** $\pm$ **1.01**) both on average *decreased* in performance, users provided feedback from **CORGI** actually *improved* with an average score difference of **1.84** $\pm$ **0.7**. While a larger sample size may be needed to observe a stronger effect (we observe $p < 0.1$ using a Welch's t-test with multiple hypothesis correction, verifying normality assumption and Medium effect size of $d = 0.52$), we provide further results showing that feedback from **CORGI** also outperforms visual feedback, and covers a diverse set of topics such as size (*"make it all a bit bigger"*) and edge straightness, in the Appendix.

Overall, our results show that **CORGI** can generate corrective feedback for novel student trajectories across a diverse set of control tasks that not only outperform baselines in automatic evaluation, but are also preferred by human raters and help learners improve at a physical control task. One appealing aspect of **CORGI** is the ability to avoid fine-tuning the underlying LM. This allows us to retrain the rich and expressive encoding the LM has learned, enabling several possible directions for future work that we discuss next.

## 5. Future Directions & Limitations

As our work is a first step towards building a model capable of generating natural language corrections for physical control tasks, there are a few limitations and opportunities for future directions. First, one important aspect of corrective feedback is tone: language with more positive encouragement can affect student learning differently, and future work could consider adding information about the student (e.g. age, personality) as an additional control for **CORGI**. Another limitation is that **CORGI** does not generate feedback with domain-specific references – future work could consider integration of corrections from **CORGI** with domain-specific approaches (Schrum et al., 2022). Additionally, while **CORGI** only provides corrections over the entire trajectory, many control tasks involve complex sequences of actions that combine many different sub-tasks, or skills. One extension of our work could consider learning how to jointly break down student trajectories into different subcomponents, and then generating corresponding feedback for each part. Finally, because **CORGI** can take any student and expert trajectory as input, potential misuse includes a malicious agent leveraging **CORGI** repeatedly to generate corrections that actually guide a student towards harmful behavior (e.g. physical actions that harm the body). An interesting avenue for future work is creating a mechanism that can detect whether an expert trajectory is plausible and safe for a human to perform under domain-specific constraints.

## 6. Acknowledgements

We thank all anonymous reviewers for their valuable feedback. We acknowledge support from Point72, Ford, AFOSR, and NSF Awards #2218760, #2132847, and #2006388. MS was also supported by the NSF GRFP under DGE-1656518.

## References


Ahn, M., Brohan, A., Brown, N., Chebotar, Y., Cortes, O., David, B., Finn, C., Gopalakrishnan, K., Hausman, K., Herzog, A., Ho, D., Hsu, J., Ibarz, J., Ichter, B., Irpan, A., Jang, E., Ruano, R. J., Jeffrey, K., Jesmonth, S., Joshi, N. J., Julian, R. C., Kalashnikov, D., Kuang, Y., Lee, K.-H., Levine, S., Lu, Y., Luu, L., Parada, C., Pastor, P., Quiambao, J., Rao, K., Rettinghouse, J., Reyes, D. M., Sermanet, P., Sievers, N., Tan, C., Toshev, A., Vanhoucke, V., Xia, F., Xiao, T., Xu, P., Xu, S., and Yan, M. Do as I can, not as I say: Grounding language in robotic affordances. *arXiv preprint arXiv:2204.01691*, 2022.

Alayrac, J.-B. and et. al., J. D. Flamingo: a visual language model for few-shot learning. *arXiv preprint arXiv:2204.14198*, 2022.

Broad, A., Arkin, J., Ratliff, N. D., Howard, T. M., and Argall, B. Real-time natural language corrections for assistive robotic manipulators. *International Journal of Robotics Research (IJRR)*, 36:684–698, 2017.

Chopra, S., Tessler, M. H., and Goodman, N. The first crank of the cultural ratchet: Learning and transmitting concepts through language. In *Cognitive Science Society*, 2019.

Cui, Y., Karamcheti, S., Palleti, R., Shivakumar, N., Liang, P., and Sadigh, D. "no, to the right" – online language corrections for robotic manipulation via shared autonomy. *arXiv preprint arXiv:2301.02555*, 2023.

Gandhi, K., Karamcheti, S., Liao, M., and Sadigh, D. Eliciting compatible demonstrations for multi-human imitation learning. In *Conference on Robot Learning (CoRL)*, 2022.

Gopalan, N., Moorman, N., Natarajan, M., Gombolay, M. C., and Georgia. Negative result for learning from demonstration: Challenges for end-users teaching robots with task and motion planning abstractions. In *Robotics: Science and Systems (RSS)*, 2022.

Hattie, J. and Timperley, H. The power of feedback. In *Review of Educational Research*, pp. 81–112, 2007.

Hayws, R. T., Jacobs, J. W., Prince, C., and Salas, E. Flight simulator training effectiveness: A meta-analysis. In *Military Psychology*, 2009.

Holtzman, A., Buys, J., Du, L., Forbes, M., and Choi, Y. The curious case of neural text degeneration. In *International Conference on Learning Representations (ICLR)*, 2020.

Ji, A. Abstract visual reasoning with tangram shapes. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

Johnson, J., Hariharan, B., van der Maaten, L., Fei-Fei, L., Zitnick, C. L., and Girshick, R. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

Lai, V. and Tan, C. On human predictions with explanations and predictions of machine learning models: A case study on deception detection. In *FAT* Conference on Fairness, Accountability, and Transparency*, 2019.

Lake, B. M., Salakhutdinov, R., and Tenenbaum, J. B. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015.

Leurent, E. An environment for autonomous driving decision-making. https://github.com/eleurent/highway-env, 2018.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pp. 740–755, 2014.

Liu, E., Stephan, M., Nie, A., Piech, C., Brunskill, E., and Finn, C. Giving feedback on interactive student programs with meta-exploration. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Lynch, C. and Sermanet, P. Language conditioned imitation learning over unstructured data. In *Robotics: Science and Systems*, 2021.

Ma, Y., Xu, G., Sun, X., Yan, M., Zhang, J., and Ji, R. X-clip: End-to-end multi-grained contrastive learning for video-text retrieval, 2022. URL https://arxiv.org/abs/2207.07285.

Maciejasz, P., Eschweiler, J., Gerlach-Hahn, K., Jansen-Troy, A., and Leonhardt, S. A survey on robotic devices for upper limb rehabilitation. In *Journal of NeuroEngineering and Rehabilitation*, 2014.

Mirchandani, S., Yu, L., Wang, M., Sinha, A., Jiang, W., Xiang, T., and Zhang, N. Fad-vlp: Fashion vision-and-language pre-training towards unified retrieval and captioning. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2022.

Patel, R. and Pavlick, E. Mapping language models to grounded conceptual spaces. In *International Conference on Learning Representations (ICLR)*, 2022.

Punnakkal, A. R., Chandrasekaran, A., Athanasiou, N., Quiros-Ramirez, A., and Black, M. J. BABEL: Bodies, action and behavior with english labels. In *Proceedings IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 722–731, June 2021.

Ruan, S. S., Jiang, L., Xu, J., Tham, B. J.-K., Qiu, Z., Zhu, Y., Murnane, E. L., Brunskill, E., and Landay, J. A. Quizbot: A dialogue-based adaptive learning system for factual knowledge. In *Conference on Human Factors in Computing Systems (CHI)*, 2019.

Schrum, M. L., Hedlund-Botti, E., and Gombolay, M. Reciprocal MIND MELD: Improving learning from demonstration via personalized, reciprocal teaching. In *6th Annual Conference on Robot Learning*, 2022. URL https://openreview.net/forum?id=f_XmiyZcsjL.

Settles, B., LaFlair, G. T., and Hagiwara, M. Machine learning–driven language assessment. In *Association for Computational Linguistics (ACL)*, 2020.

Sharma, P., Sundaralingam, B., Blukis, V., Paxton, C., Hermans, T., Torralba, A., Andreas, J., and Fox, D. Correcting robot plans with natural language feedback. In *Robotics: Science and Systems (RSS)*, 2022.

Srivastava, M., Biyik, E., Mirchandani, S., Goodman, N., and Sadigh, D. Assistive teaching of motor control tasks to humans. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Sumers, T. R., Ho, M. K., and Griffiths, T. L. Show or tell? demonstration is more robust to changes in shared perception than explanation. In *Cognitive Science Society*, 2020.

Tevet, G., Gordon, B., Hertz, A., Bermano, A. H., and Cohen-Or, D. MotionCLIP: Exposing human motion generation to CLIP space. In *European Conference on Computer Vision*, 2022.

Tsimpoukelli, M., Menick, J., Cabi, S., Eslami, S. M. A., Vinyals, O., and Hill, F. Multimodal few-shot learning with frozen language models. *arXiv preprint arXiv:2204.14198*, 2021.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., and Brew, J. HuggingFace's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019a.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Scao, T. L., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. M. Huggingface's transformers: State-of-the-art natural language processing, 2019b. URL https://arxiv.org/abs/1910.03771.

Wong, C., Ellis, K., Tenenbaum, J. B., and Andreas, J. Leveraging language to learn program abstractions and search heuristics. In *International Conference on Machine Learning (ICML)*, 2022.

Yi, Y., Deng, H., and Hu, J. Improving image captioning evaluation by considering inter references variance. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 985–994, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.93. URL https://aclanthology.org/2020.acl-main.93.

Yu, C., Xu, Y., Li, L., and Hsu, D. Coach: Cooperative robot teaching. In *Conference on Robot Learning (CoRL)*, 2022.

Zhong, R., Snell, C., Klein, D., and Steinhardt, J. Describing differences between text distributions with natural language, 2022. URL https://arxiv.org/abs/2201.12323.

# A. Appendix

We include information about accessing our dataset, model checkpoints, and user study infrastructure at this link: https://github.com/Stanford-ILIAD/corgi.

## A.1. Ethics Statement & IRB

The purpose of our work is to help student learners improve performance on control tasks by automatically generating fluent and accuracy feedback in natural language. However, because physical control tasks can affect user comfort and health, an important risk of our work is its potential to mislead a person to perform control movements that may be harmful. Furthermore, a malicious actor can leverage the method behind **CORGI** to train a model that intentionally hurts user performance. For these reasons, we emphasize the importance of ensuring safety checks when deploying a system based on **CORGI** and exercising caution in critical application areas.

Human subject studies, including both the human preference and learning performance evaluations, were conducted as part of a study approved by Stanford University's Institutional Review Board (protocol # IRB-49406). Participants were asked to agree to a consent form (like this example), before continuing to the study interface. All participants were crowdworkers recruited on the Prolific platform.

## A.2. Expert Trajectory Assumptions

One important assumption of our work is the need for an expert reference trajectory used to provide feedback. While all experiments in this work are conducted with a limited range of experts, in reality there exist multiple expert behaviors for a task (e.g. using the right hand or the left hand) that result in different trajectories. An ideal teaching system would be able to take as input any arbitrary expert behavior, and provide appropriate corrective feedback for the system. While **CORGI** has this capability with respect to its API (any arbitrary expert behavior can be sent as input), we chose not to cover an exhaustive range of expert behavior due to nuance in defining different "optimal" experts: for example, in the DRAWING task, while drawing the letter "I" bottom-up or top-down might be equally optimal, this may not be true for particular applications like rehabilitation, where a trained may seek to guide a student towards a specific expert behavior. Furthermore, we believe one important aspect of good teaching is developing strong priors on the types of mistakes a student might make for a given task. For example, before even observing a student, a tennis instructor may know that hitting a ball too low is a common mistake. Training a model over a selected set of expert references, rather than across any possible trajectory as an expert, can help provide this inductive bias. Nevertheless, we introduce variance in expert trajectories for each task by (i) varying characters for DRAWING, (ii) perturbations to expert trajectories in STEERING, and (iii) multiple expert demonstrations for MOVEMENT. Future work could consider training on more varied expert behavior as well as designing a system to identify which expert behavior to provide as input to **CORGI**, depending on the student's learning preferences.

## A.3. Training Details

The trajectory encoder $\mathcal{M}_{\text{traj},\theta}$ part of **CORGI** is trained for 200 epochs on one NVIDIA A40 GPU with a batch size of 64 and learning rate of 0.05, although we observed little sensitivity in performance with respect to learning rate. We split our training dataset into train and valid splits, and use the latter to perform early stopping. We repeat the same training procedure for both model ablations (no pre-training and no data-augmentation). The frozen LM we use is the 124M-parameter version of GPT-2 from Wolf et al. (2019a).

We set the parameter $n$ for $\mathcal{M}_{\text{traj},\theta}$ to be 20, so the trajectory encoder outputs a set of 20 vectors with dimension 768. $\mathcal{M}_{\text{traj},\theta}$ is a 3-layer feed-foward neural network, where each layer has an output size of $n = 20 \times 768$.

For the STEERING task, we use trajectories from partially-trained Soft Actor-Critic agents trained for only 100 epochs using the StableBaselines3 implementation as some of our student trajectories. This leads to a variety of failure modes, which we human annotators describes.

## A.4. Crowdsourcing Language Corrections

For each of our three control tasks, we recruit crowdworkers on Prolific to provide corrective feedback to a student given pairwise student and expert trajectories, as seen in Figure 4. Each crowdworker provides 10 language corrections, and we pay then 14 USD per hour. In total, we collect **2,023** corrections.

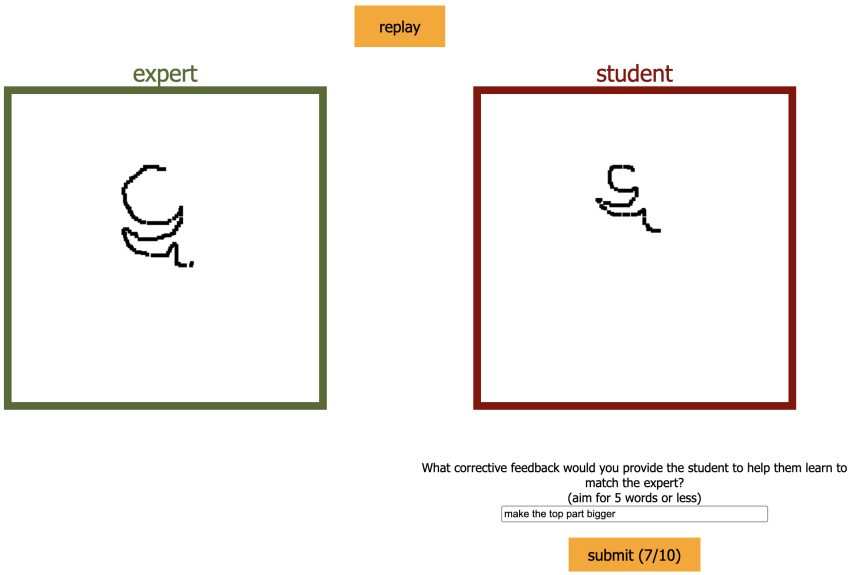


*Figure 4.* User Interface for Crowdsourcing Corrections for the Drawing task

### A.5. Human Preference Evaluation

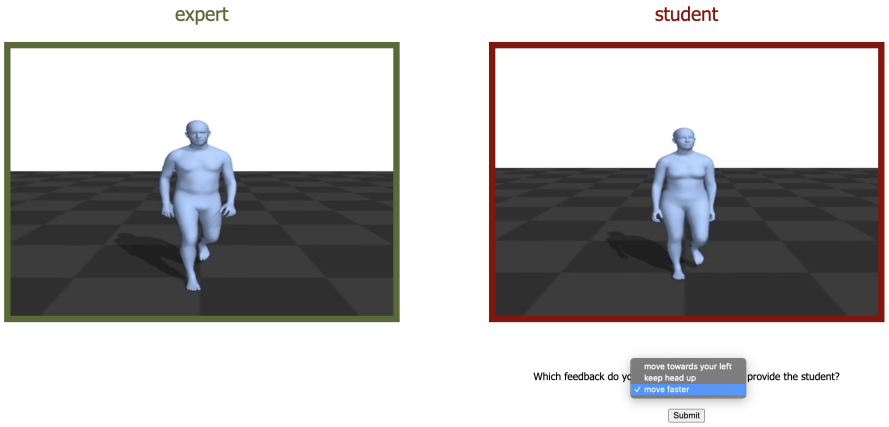


*Figure 5.* User Interface of the Human Preference Evaluation

For each of our three control tasks, we recruit crowdworkers on Prolific to select their preferred feedback to provide to a student given pairwise student and expert trajectories and a dropdown list of language corrections to pick from, as seen in Figure 5. Each crowdworker provides 10 preferences, and we pay then 14 USD per hour. In total, we collect **1,800** preference ratings.

**Direct Pairwise Comparisons** In addition to our main results, we use the same interface to conduct direct pairwise comparisons from participant preferences between feedback from CORGI vs. **Nearest Neighbors** and feedback from CORGI vs. **Ground Truth**. We report these results in Table 3, which support the results reported in the main paper: CORGI outperforms the **Nearest Neighbor** baselines significantly for WRITING and STEERING tasks, and even outperforms **Ground Truth** annotations for the STEERING task.

### A.6. Human Learning Evaluation

We evaluate the degree corrections from CORGI help humans learn for the DRAWING task by recruiting 60 crowdworkers on Prolific, split evenly between two control groups (random feedback and no feedback) and the experiment group, to try

*Table 3.* Users are significantly more likely to prefer **CORGI** over **Nearest Neighbors** for the WRITING and STEERING tasks, and even outperforms Ground Truth feedback for the STEERING task. Asterisk (*) marks results that are statistically significant ($p < 0.05$) with multiple hypothesis correction, using a binomial test where the null hypothesis is set to equal preference rate.

| DOMAIN | % CORGI PREFERRED VS. NEAREST NEIGHBORS | % CORGI PREFERRED VS. GROUND TRUTH |
|---|---|---|
| WRITING | $74 \pm 4.1^*$ | $58 \pm 5.6$ |
| STEERING | $59 \pm 3.7^*$ | $60 \pm 3.3^*$ |
| MOVEMENT | $54 \pm 3.1$ | $45 \pm 2.5$ |

drawing a target stimulus as seen in Figure 6. Each crowdworker provides three drawing trajectories, and we measure the difference between the third and first trial in terms of error with respect to the (hidden) expert trajectory. We pay each crowdworker 14 USD per hour. Example user trajectories can be seen in Figure 7.

Feedback generated from **CORGI** covers a diverse set of topics for participants in our user study. While find that $70\%$ of corrections focus on size (split evenly between increasing and decreasing size), several participants received feedback about line sharpness (e.g. 13%) and straightness (10%). Additionally, there was a long tail of corrections that were only generated once for a student (e.g. "make it stronger", referring to the drawing line weight). Even for corrections referring to size, there exists variation in the degree of the correction (e.g. "needs to be a bit larger" vs. "make it smaller").

**Visual Feedback Comparison** Finally, we run an additional experiment evaluating providing visual feedback, instead of language feedback, by providing a visual overlay on the drawing canvas. This naturally makes the task easier for more stationary environments like drawing. However, observations from a user study conducted don 20 additional crowdworkers recruited on Prolific show that while indeed participants perform on average around **10.1** points (between 0 and 100) higher in overall task performance than students receiving language feedback from CORGI, the learning gain (change in error from expert trajectory) is **0.39 +/- 0.48**, which is lower than those provided language feedback from CORGI. This is likely because learners, when given access to a visual overlay for this task, can immediately start to perform well, while language identifying specific areas to improve on can be remembered long-term by students.

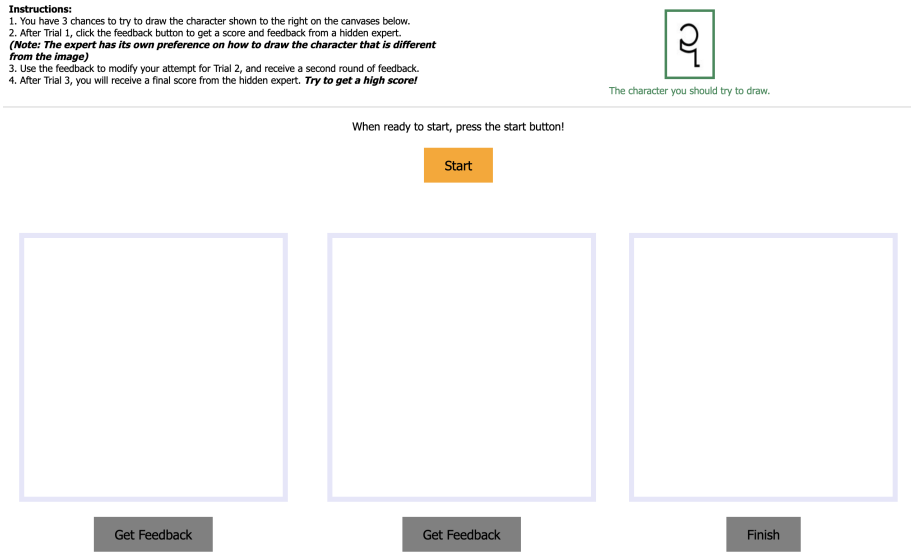


*Figure 6.* User Interface of the Human Learning Evaluation

*Figure 7.* Example User trajectories with feedback from our model