# Progressively Efficient Learning

*Ruijie Zheng[♢], *Khanh Nguyen[♣], Hal Daumé III[♢♡], Furong Huang[♢], Karthik Narasimhan[♠]
[♢] University of Maryland, College Park [♡] Microsoft Research
[♣] University of California, Berkeley [♠] Princeton University

## Abstract

Assistant AI agents should be capable of rapidly acquiring novel skills and adapting to new user preferences. Traditional frameworks like imitation learning and reinforcement learning do not facilitate this capability because they support only low-level, inefficient forms of communication. In contrast, humans communicate with *progressive efficiency* by defining and sharing abstract intentions. Reproducing similar capability in AI agents, we develop a novel learning framework named ***C**ommunication-**E**fficient **I**nteractive **L**earning* (CEIL). By equipping a learning agent with an abstract, dynamic language and an intrinsic motivation to learn with minimal communication effort, CEIL leads to emergence of a human-like pattern where the learner and the teacher communicate progressively efficiently by exchanging increasingly more abstract intentions. CEIL demonstrates impressive performance and communication efficiency in a 2D MineCraft domain featuring long-horizon decision-making tasks. Agents trained with CEIL quickly master new tasks, outperforming non-hierarchical and hierarchical imitation learning by up to 50% and 20% in absolute success rate, respectively, given the same number of interactions with the teacher. Especially, the framework performs robustly with teachers modeled after human pragmatic communication behavior.

## 1 Introduction

Imagine Alice, a programming expert, teaching Bob, a novice, how to write computer programs. Initially, because they share little common knowledge in this domain, Alice has to demonstrate step by step how a program is written. This strategy quickly enables Bob to write simple programs, but it is inadequate for teaching him to compose sophisticated programs consisting of thousands of lines of code. Hence, after teaching through demonstrations for a while, Alice switches to a more efficient strategy: she grows a shared vocabulary with Bob and gradually adds to it increasingly more abstract terms that help them express complex intentions succinctly. For example, after explaining the concepts of "*for-loop*" and "*a function that checks whether an integer is a prime*", Alice can teach Bob to count the number of two-digit primes by giving a high-level instruction like "*write a* for loop *from 1 to 99, and call the* prime-checking function *in each iteration*" rather than having to dictate a full program to him. In general, as Alice and Bob communicate more frequently and want to exchange increasingly intricate ideas, they reduce effort by making their communication more abstract. We refer to this phenomenon as *progressively efficient communication*.

In order to excel as personal assistants of humans, AI agents should be capable of progressively efficient communication. These agents should handle increasingly complex user requests without demanding extensive user effort to adapt their behavior. In this paper, we demonstrate that incorporating elements of human communication allows for the construction of such agents. We first identify two elements that are prerequisite for progressively efficient communication but are missing in traditional frameworks like imitation learning (IL) and reinforcement learning (RL): (i) a dynamic, referential

*The first two authors contribute equally. Correspondence email: `kxnguyen@berkeley.edu`.

Workshop on Intrinsically Motivated Open-ended Learning (NeurIPS 2023).

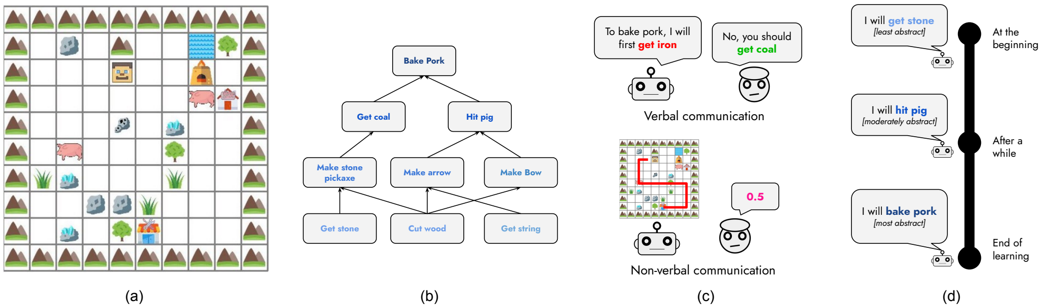


Figure 1: An illustration of CEIL on a "bake pork" task in a 2D MineCraft environment (a). The environment features compositional tasks which represent intentions at various levels of abstraction (b). CEIL enables the learner to quickly learn complex tasks by capitalizing on its mastery of simpler tasks. When interacting with the teacher (c), the learner can choose to verbally communicate an intention (a reference to an action sequence) and receive correction, or execute an intention by taking actions in the environment and obtain evaluation. It can alternate between these two modes of communication within a learning episode. Aiming to reduce communication effort, the learner conveys increasingly more abstract intentions over time (d). Learning efficiency enhances as communication becomes more abstract.

communication medium (the means) and (ii) a desire to minimize collaborative effort (the motivation). We develop ***C**ommunication-**E**fficient **I**nteractive **L**earning* (CEIL), a learning framework that equips the learning agent with human-like means and motivation for progressively efficient communication. As illustrated in Figure 1, CEIL transcends IL and RL by allowing the teacher and the learner to exchange abstract intentions rather than low-level signals like numerical rewards or primitive actions. Furthermore, it injects into the learner an intrinsic motivation to minimize long-term communication effort, encouraging the learner to understand and use abstract terms to express intentions concisely. While incorporating one of these elements of human communication has been previously explored [Kulkarni et al., 2016, Le et al., 2018, Ren et al., 2021, Brantley et al., 2020, Zhang and Cho, 2016], our work is the first to integrate *both* of them in a single framework with the goal of mimicking the progressive efficiency of human communication.

We present a variant of our framework with IL-like instructive feedback and RL-like evaluative feedback, and propose a practical learning algorithm by extending Q-learning. We evaluate the effectiveness of our algorithm on a challenging 2D MineCraft domain where each task is composed of various subtasks and requires a long sequence of actions to complete. Results indicate that our algorithm CEIL learns significantly faster and achieves higher asymptotic performance than various RL and IL baselines. The algorithm achieves performance gain consistently across various models of the teacher we construct, including those mimicking human pragmatic behavior. The temporal change in the distribution of utterances indicates that progressively efficient communication indeed emerges, as the learner conveys increasingly more abstract intentions to the teacher. Our work illustrates that integrating human communication traits holds great promise in the advancement of more efficient and human-compatible learning frameworks.

## 2 Overview

In this section, we highlight the key novelties of our framework. More details of the implementation are provided in Appendix A.

**Referential, productive communication with abstract intentions.** CEIL allows the learner and the teacher to convey *abstract intentions*. An intention is a symbol that refers to a sequence of actions that aims to accomplish a task (e.g., "bake pork", "make arrow"). Using intention as the medium, CEIL supports referential communication. Moreover, the framework allows for the expansion of the set of intentions and the composition of existing intentions for expressing new intentions, making communication also productive.

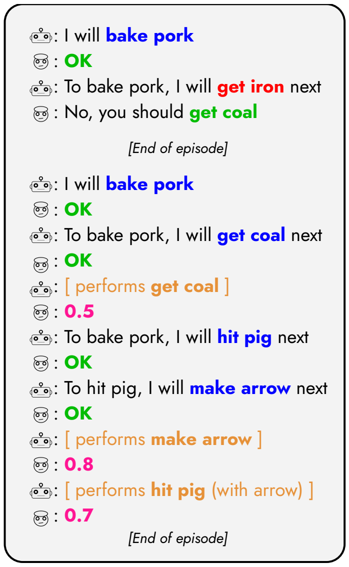
We frame learning in CEIL as a communicative activity, where the goal is for the learner and the teacher to agree on the meaning of the intention referring to the main task. To achieve this goal, the two interlocutors repeat a process in which the learner conveys its current interpretation of the intention, and the teacher provides feedback to rectify that interpretation. To convey its interpretation of an intention, the learner can select between two general options:

(a) **verbal communication**: express the intention in terms of other intentions;
(b) **non-verbal communication**: execute the intention by taking actions in the environment to perform the task that it refers to.

For option (a), in our implementation, the learner utters only an initial part of the expression so that the teacher can correct it more efficiently. For example, when learning a "bake pork" task, it would say "*[to bake pork, I will] get coal*" rather than "*[to bake pork, I will] get coal, then hit pig*".

Upon observing the learner's interpretation, the teacher offers feedback. CEIL can be instantiated with various types of feedback. In this paper, we present a variant in which the teacher issues IL-like *instructive feedback* and RL-like *evaluative feedback*. Specifically, if the learner chooses option (a), the teacher provides instructive feedback in the form of intention correction. When the proposed intention is correct, the teacher simply confirms. If the learner chooses option (b), the teacher offers evaluative feedback in the form of numerical evaluation of task execution. With this option, other types of feedback, such as language descriptions [Nguyen et al., 2021] can be incorporated. We choose numerical feedback to simplify the learning algorithm.

Figure 2 shows example conversations between the learner and the teacher. Since our focus is on communication at the intention level, we simplify the problems of generating natural expressions to convey intentions and interpreting intentions from natural expressions. Despite the uncomplicated look of the language, our communication protocol is highly general, since the learner and the teacher can potentially exchange a broad set of intentions. In particular, the protocol strictly generalizes those of IL and RL: (hierarchical) IL is equivalent to CEIL with only verbal communication, while RL (with sparse reward) is analogous to CEIL with only non-verbal communication.



🤖: I will **bake pork**
🙂: **OK**
🤖: To bake pork, I will **get iron** next
🙂: No, you should **get coal**

*[End of episode]*

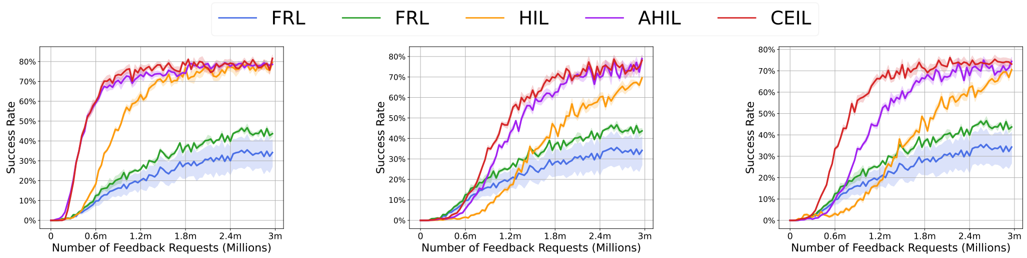
🤖: I will **bake pork**
🙂: **OK**
🤖: To bake pork, I will **get coal** next
🙂: **OK**
🤖: [ performs **get coal** ]
🙂: **0.5**
🤖: To bake pork, I will **hit pig** next
🙂: **OK**
🤖: To hit pig, I will **make arrow** next
🙂: **OK**
🤖: [ performs **make arrow** ]
🙂: **0.8**
🤖: [ performs **hit pig** (with arrow) ]
🙂: **0.7**

*[End of episode]*

Figure 2: Conversations between the learner and teacher for learning the "bake pork" task.

**Pragmatic communication to save long-term effort.** Within the realm of verbal communication, CEIL allows interlocutors to convey expressions of varying levels of abstraction. We adopt a simple notion of level of abstraction, defining it as the number of actions that an intention refers to. For example, in CEIL, the learner may utter either "*[to bake pork I will] make stone pickaxe*" or "*[to bake pork I will] get stone*" to refer to an initial step of "bake pork", but the former is a more abstract expression because "get stone" is a subtask of "make stone pickaxe".

Having flexibility in the language enables the learner to communicate pragmatically to optimize for a goal. In CEIL, the learner's goal is to minimize the long-term (joint) communication effort. We express this goal by setting a learning objective that minimizes short-term communication effort while also minimizing task error. Striving for this objective drives the learner to speak with the teacher at a level of abstraction that best suits their current mutual knowledge and to gradually speak more abstractly. Specifically, the learner begins by communicating using the least abstract intentions, as the meanings of those are easiest to learn. At this stage, the communication effort is substantial: the learner proposes numerous intentions during an episode, inducing tantamount effort from the teacher to provide feedback. CEIL allows the learner to explore, occasionally uttering more abstract intentions. The urge to save communication effort prompts the learner to improve its understanding of these abstract intentions (by incorporating the teacher's feedback) and to leverage them incrementally more often to shorten its verbal expressions. When the learner has mastered the main task, it will communicate minimally with the teacher. It will simply declare the intention of performing the task (e.g., "*[I will] bake pork*") and execute that intention immediately, which is the same as its behavior at test time.



(a) Top-down (non-pragmatic) teacher

(b) Language-based pragmatic teacher

(c) Performance-based pragmatic teacher

Figure 3: Success rate on the training task (BakePork) as a function of number of feedback requests made to the teacher. Results are averaged over four random seeds.

It is important to emphasize that CEIL does *not* directly force the learner to speak increasingly more abstractly. Much like in humans, this capability emerges as a means for the learner to achieve its *socially motivated goal*—to communicate effectively with minimal effort.

**Pragmatic teachers.** Another novelty of our framework is the employment of teachers modeled after human pragmatic behavior. Hierarchical IL or RL (e.g., [Le et al., 2018]) typically assumes a *top-down* teacher, who always recommends the most abstract intention to correct the learner's intention. Consider the example in Figure 2, where the learner falsely proclaims "*[to bake pork I will] get iron*". According to the task tree in Figure 1, "get coal", "make stone pickaxe", "get stone" are all valid intentions to refer to the learner. A top-down teacher would choose "get coal", the most abstract, to utter. This teacher is non-pragmatic because it ignores the learner's behavior. To better mimic communication with humans, we simulate two types of pragmatic teacher, each of which employs a heuristic to select an intention that is deemed easiest for the learner to interpret. The *language-based* teacher replies with the intention whose level of abstraction is most similar to that of the proposed intention of the learner. Meanwhile, the *performance-based* teacher samples an intention among the candidates with probability proportional to the historical execution success rate of the learner. More details about these teachers are given in §A.7.

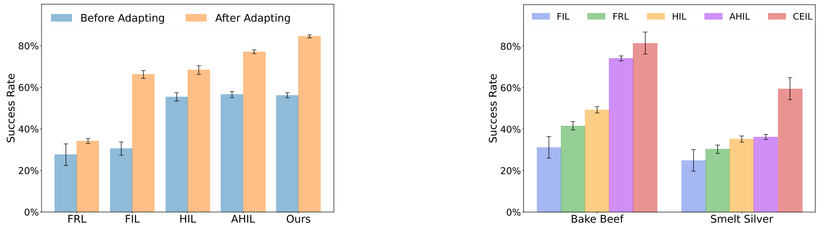
Unlike the top-down teacher, the two pragmatic teachers lack motivation to communicate abstractly. We will empirically demonstrate that regardless of whether the teacher possesses this motivation, our learner is still able to drive communication to be incrementally more abstract and efficient.

## 3 Experimental Setup

**Environment.** We employ the simulator developed by [Sohn et al., 2018], which emulates a MineCraft-style game on an $8 \times 8$ grid. The player can move, interact with other entities to collect items, and combine them to create new items.

**Baselines.** We compare with: (a) *flat imitation learning* (FIL) implements DAgger [Ross et al., 2011], (b) *flat reinforcement learning* (FRL) is standard Q-learning, (c) *hierarchical imitation learning* (HIL) performs DAgger with a teacher that can suggest high-level intentions, and (d) active hierarchical imitation learning (AHIL) is an ablated version of our algorithm which intrinsic motivation concerns only performance maximization.

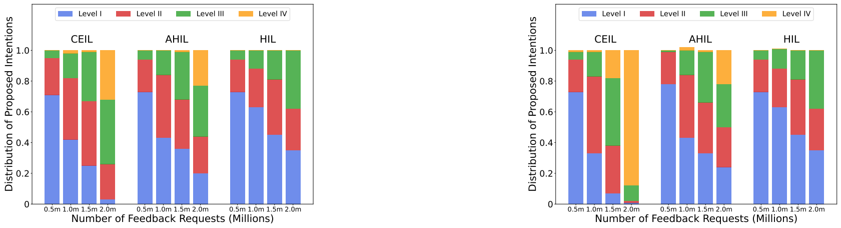
**Training settings.** We evaluate all approaches on three settings. In the *learn-from-scratch* setting, agents are trained from random parameter initialization to perform the BakePork task, which requires on average 68 actions to complete. In the *environment-adaptation* setting, we put the agents learned in the first setting in new environment layouts and continue training them on the BakePork task. In the *task-adaptation* setting, we instead train those agents to perform two novel tasks, BakeBeef and SmeltSilver. These two tasks share several common subtasks with the BakePork task. All agents are given a budget of 10k feedback requests for the two adaptation settings.



(a) Adaptation to new environment layouts

(b) Adaptation to new tasks

Figure 4: Performance of different approaches when adapted to new BakePork environments and to new tasks (BakeBeef and SmeltSilver). All agents learn with the performance-based pragmatic teacher. They were pre-trained on the BakePork task to reach the same success rate, and then were adapted with the same request budget of 10K. In both settings, CEIL adapts more successfully, showing that the mutual knowledge it acquired with the teacher during the learn-from-scratch phase generalizes more robustly.



(a) Language-based pragmatic teacher

(b) Performance-based pragmatic teacher

Figure 5: Distribution of uttered intentions on the BakePork task. We group the intentions into four levels of abstraction, which is a function of the distance to the root in the task tree (level IV is most abstract). CEIL exhibits a strong inclination to propose increasingly more abstract intentions.

## 4 Results

**Learn-from-scratch setting.** In Figure 3, we plot the performance of CEIL and the baselines against the number of teacher feedback requests. CEIL outperforms all baselines in terms of both asymptotic success rate and sample efficiency. With the non-pragmatic teacher, CEIL reaches the same final performance as hierarchical IL approaches, but it learns much faster. With the pragmatic teachers, CEIL not only learns faster, but also achieves a higher final success rate.

**Adaptation settings.** Figure 4 shows the performance of the agents in the two adaptation settings. Interestingly, while CEIL, HIL, and AHIL were pre-trained to have the same performance on the BakePork task, their adaptation performances differ significantly. CEIL attains the highest success rate in all settings. Most vividly, in adapting to the SmeltSilver task, the CEIL outperforms HIL and AHIL by a subtantial margin of more than 20% in absolute success rate. This shows that CEIL induces a communication capability that generalizes much better than the other two approaches. The gaps between CEIL and other methods are smaller when adapting to the BakeBeef task or to new BakePork environments. The generalization challenge in these scenarios is not as significant as in learning the SmeltSilver task: the agents mostly need to further improve learned skills, while they have to learn many novel skills to smelt silver. These results show that our framework not only induces efficient learning but also enables robust generalization. This is somewhat surprising, because the intrinsic motivation we install in the agent does not explicitly force it to generalize better.

**Does the agent communicate more abstractly over time?** To answer this question, we visualize the distribution of intentions proposed by the agent over time. We divide the intentions into four groups, from least to most abstract. Figure 5 shows the changes in the way agents communicate with the two pragmatic teachers. We observe that AHIL and HIL also enables the agent to speak more abstractly over time. However, CEIL induces the strongest manifestation of this phenomenon.

## References

Tejas D Kulkarni, Karthik Narasimhan, Ardavan Saeedi, and Josh Tenenbaum. Hierarchical deep reinforcement learning: Integrating temporal abstraction and intrinsic motivation. *Advances in neural information processing systems*, 29, 2016.

Hoang Le, Nan Jiang, Alekh Agarwal, Miroslav Dudík, Yisong Yue, and Hal Daumé III. Hierarchical imitation and reinforcement learning. In *International conference on machine learning*, pages 2917–2926. PMLR, 2018.

Pengzhen Ren, Yun Xiao, Xiaojun Chang, Po-Yao Huang, Zhihui Li, Brij B Gupta, Xiaojiang Chen, and Xin Wang. A survey of deep active learning. *ACM computing surveys (CSUR)*, 54(9):1–40, 2021.

Kianté Brantley, Amr Sharaf, and Hal Daumé III. Active imitation learning with noisy guidance. *arXiv preprint arXiv:2005.12801*, 2020.

Jiakai Zhang and Kyunghyun Cho. Query-efficient imitation learning for end-to-end autonomous driving. *arXiv preprint arXiv:1605.06450*, 2016.

Khanh X Nguyen, Dipendra Misra, Robert Schapire, Miroslav Dudík, and Patrick Shafto. Interactive learning from activity description. In *International Conference on Machine Learning*, pages 8096–8108. PMLR, 2021.

Sungryull Sohn, Junhyuk Oh, and Honglak Lee. Hierarchical reinforcement learning for zero-shot generalization with subtask dependencies. *Advances in neural information processing systems*, 31, 2018.

Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635. JMLR Workshop and Conference Proceedings, 2011.

He He, Jason Eisner, and Hal Daume. Imitation learning by coaching. *Advances in neural information processing systems*, 25, 2012.

Sanmit Narvekar, Bei Peng, Matteo Leonetti, Jivko Sinapov, Matthew E Taylor, and Peter Stone. Curriculum learning for reinforcement learning domains: A framework and survey. *The Journal of Machine Learning Research*, 21(1):7382–7431, 2020.

Gaurav Kumar, George Foster, Colin Cherry, and Maxim Krikun. Reinforcement learning based curriculum optimization for neural machine translation. *arXiv preprint arXiv:1903.00041*, 2019.

Mingjun Zhao, Haijiang Wu, Di Niu, and Xiaoli Wang. Reinforced curriculum learning on pre-trained neural machine translation models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9652–9659, 2020.

Carlos Florensa, David Held, Markus Wulfmeier, Michael Zhang, and Pieter Abbeel. Reverse curriculum generation for reinforcement learning. In *Conference on robot learning*, pages 482–495. PMLR, 2017.

Minghuan Liu, Hanye Zhao, Zhengyu Yang, Jian Shen, Weinan Zhang, Li Zhao, and Tie-Yan Liu. Curriculum offline imitating learning. *Advances in Neural Information Processing Systems*, 34: 6266–6277, 2021.

Richard S Sutton, Doina Precup, and Satinder Singh. Between mdps and semi-mdps: A framework for temporal abstraction in reinforcement learning. *Artificial intelligence*, 112(1-2):181–211, 1999.

Burr Settles. Active learning literature survey. 2009.

Daniel Hsu. A new framework for query efficient active imitation learning. *arXiv preprint arXiv:1912.13037*, 2019.

Kshitij Judah, Alan Paul Fern, and Thomas Glenn Dietterich. Active imitation learning via reduction to iid active learning. In *2012 AAAI Fall Symposium Series*, 2012.

Lisa Torrey and Matthew Taylor. Teaching on a budget: Agents advising agents in reinforcement learning. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pages 1053–1060, 2013.

Felipe Leno Da Silva, Pablo Hernandez-Leal, Bilal Kartal, and Matthew E Taylor. Uncertainty-aware action advising for deep reinforcement learning agents. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 5792–5799, 2020.

Aron Culotta and Andrew McCallum. Reducing labeling effort for structured prediction tasks. In *AAAI*, volume 5, pages 746–751, 2005.

Khanh Nguyen, Debadeepta Dey, Chris Brockett, and Bill Dolan. Vision-based navigation with language-based assistance via imitation learning with indirect intervention. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12527–12537, 2019.

Khanh Nguyen and Hal Daumé III. Help, anna! visual navigation with natural multimodal assistance via retrospective curiosity-encouraging imitation learning. *arXiv preprint arXiv:1909.01871*, 2019.

Meng Fang, Yuan Li, and Trevor Cohn. Learning how to active learn: A deep reinforcement learning approach. *arXiv preprint arXiv:1708.02383*, 2017.

Khanh X Nguyen, Yonatan Bisk, and Hal Daumé Iii. A framework for learning to request rich and contextually useful information from humans. In *International Conference on Machine Learning*, pages 16553–16568. PMLR, 2022.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*, 2022.

OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*, 2022.

Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021. URL `http://jmlr.org/papers/v22/20-1364.html`.

# A Appendix

## A.1 Related work

Previous work on curriculum-based IL and RL [He et al., 2012, Narvekar et al., 2020, Kumar et al., 2019, Zhao et al., 2020, Florensa et al., 2017, Liu et al., 2021] has extended the pragmatic communication capabilities of the teacher by enabling them to choose the order of tasks to teach the learner. These strategies can be viewed as making the teacher more pragmatic in choosing *which* intentions to convey. But in choosing *how* to express those intentions, the teacher is restricted to a low-level language. Work on hierarchical IL and RL [Kulkarni et al., 2016, Sutton et al., 1999, Le et al., 2018] attempts to enrich the language of the teacher and the learner, allowing them to exchange high-level intentions. This provides a better means for efficient communication, but the learner still follows the teacher's way of communication without having its own intrinsic motivation to communicate more efficiently over time. Moreover, most formulations adopt a hierarchy of only two levels, constraining the flexibility of the language.

Active learning [Ren et al., 2021, Settles, 2009] aims to reduce the feedback queries made to the teacher. This framework has been instantiated in non-hierarchical IL and RL settings [Hsu, 2019, Brantley et al., 2020, Judah et al., 2012, Torrey and Taylor, 2013]. Our work shows that combining active learning with a rich, flexible language results in much more efficient learning. Previous active learning strategies are based on intrinsic uncertainty [Da Silva et al., 2020, Culotta and McCallum, 2005, Nguyen et al., 2019], error prediction [Zhang and Cho, 2016, Nguyen and Daumé III, 2019], or direct optimization of the number of queries via reinforcement learning [Fang et al., 2017, Nguyen et al., 2022]. We demonstrate that minimizing both the number of queries and the task error is important for progressively efficient communication to emerge strongly.

Recent advances in large language models allow humans to teach them through highly natural language [Brown et al., 2020, Chowdhery et al., 2022, OpenAI, 2023]. Users can alternate the behavior of these systems using complex instructions and in-context examples [Bubeck et al., 2023, Wei et al., 2022]. However, due to lacking theoretical guarantees, it remains unclear how to reliably inject intrinsic motivation into these models. Our work focuses on learning through parameter optimization, which allows us to easily adapt the model continually and enforce intrinsic motivation.

## A.2 Enriching the means: communication of referential intentions

**Intentions.** Let $\mathcal{I} \subseteq \mathcal{G}$ be the set of intentions the learner can convey. Initially, it contains a set of seed intentions, which includes the intention of performing the main task. The seed intentions can be constructed with a knowledge base. For example, one can collect a comprehensive list of MineCraft tasks from the game's Wiki. Moreover, this set is *expandable*: when the teacher introduces a new task to the learner, it can add to set the task's name as a new intention. We do not specify the levels of abstraction of the intentions to the learner. It realizes this quality through interaction with the teacher.

**Learner components.** The learner has two components: a *policy* $\pi_\theta(u \mid s, i)$ and a *memory* $M(u)$. The policy takes as input an environment state $s \in \mathcal{S}$ and an intention $i \in \mathcal{I}$, and outputs a distribution over actions $u \in \mathcal{I} \cup \{\texttt{[do]}, \texttt{[done]}\}$, where `[do]` and `[done]` are special actions which we will define shortly. The memory helps the learner update and keep track of its current intention. Each action $u$ can be verbal or executive. Taking a verbal action ($u \in \mathcal{I} \cup \{\texttt{[done]}\}$) alters the current intention of the learner (its mental state), while taking an executive action ($u = \texttt{[do]}$) changes its environment state.

**Interactions during an episode (Algorithm 1).** The learner starts in state $s_1$ and conveys the intention $i_1 = g$, which is to perform the main task. At the time step $t$, suppose the learner's intention and environment state are $i_t$ and $s_t$, respectively. The learner selects an action $u_t \sim \pi_\theta(s_t, i_t)$. If $u_t \in \mathcal{I} \cup \{\texttt{[done]}\}$, the learner chooses to verbally communicate an intention. The `[done]` action represents the intention of relinquishing the current intention. The learner computes a new current intention $i_{t+1} = M(u_t)$ by querying the memory with the action. Meanwhile, the environment state remains the same $s_{t+1} = s_t$. After that, the learner receives instructive feedback $f_t = u_t^\star \in \mathcal{I} \cup \{\texttt{[done]}\}$ from the teacher, which indicates the correct intention in the current state.

If $u_t = \texttt{[do]}$, the learner elects to execute the current intention $i_t$. In this case, the agent continuously taking primitive actions until it decides to terminate, generating an execution

$(s_t^1 = s_t, a_t^1, \cdots, s_t^L, a_t^L)$, where $L$ is the trajectory length, $a_t^l \in \mathcal{A}$ for $1 \leq l \leq L$. It then updates its current intention and environment state, setting $i_{t+1} = M(\texttt{[done]})$ and $s_{t+1} = s_t^L$, and receives evaluative feedback from teacher, which is a score $f_t \in \mathbb{R}$ judging the execution.

**Memory.** Following Nguyen et al. [2022], we implement the memory $M$ as a stack data structure, which prioritizes the most recently proposed intentions. Initially, the stack contains only the main-task intention $i_1$. When the learner chooses to communicate verbally, the communicated intention is pushed to the stack. However, when the special `[done]` action is taken, the intention at the top of the stack is popped. After pushing or popping, the intention at the top of the stack is returned as the current intention. If there is no intention in the stack to return, the episode ends. This stack-based memory allows the learner to learn a deep hierarchy of tasks, making CEIL more general than prior work on hierarchical IL and RL that assumes only a two-level hierarchy [Kulkarni et al., 2016, Le et al., 2018]. We defer the exploration of more intricate memory designs, such as interleaving executions of multiple tasks or prioritizing tasks that cost less resources to execute from the current state.

**Algorithm 1** CEIL training episode

```
Observe initial state $s_1$ and main task $g$
Set initial intention $i_1 = g$
Initialize memory stack $M = \{i_1\}$
$t = 0$
while not $M$.empty() do
    $t \leftarrow t + 1$
    Get current intention $i_t = M$.top()
    Choose action $u_t \sim \pi_\theta(s_t, i_t)$
    if $u_t \in \mathcal{I} \cup \{\texttt{[done]}\}$ then    // verbal
        Receive instructive feedback $f_t = u_t^\star$
        Stay in the same state $s_{t+1} = s_t$
        Set new intention $i_{t+1} = M(u_t)$
    else    // non-verbal
        Execute $i_t$ and arrive in final state $s_t^L$
        Receive evaluative feedback $f_t \in \mathbb{R}$
        Set new state $s_{t+1} = s_t^L$
        Set new intention $i_{t+1} = M(\texttt{[done]})$
    end if
end while
Update policy $\pi_\theta$ w.r.t. learning objective
```

### A.3 Injecting the motivation: minimization of long-term communication effort

**Learner's intrinsic motivation.** As mentioned, the CEIL learner aims to minimize the long-term communication effort. To formalize this goal, let $\pi_0$ be the learner's current policy and $\pi_n$ be the its policy after $n$ learning updates. For simplicity, we assume the policy is updated after every learning episode. We associate a communication cost $c(s, i, u)$ with every action $u$ taken by the learner when it is in state $s$ with intention $i$. $C(\tau) = \sum_{t=1}^{T} c(s_t, i_t, u_t)$ represents the communication effort in an episode an episode $\tau = \{(s_t, i_t, u_t)\}_{t=1}^{T}$. Let the *task error* $J_{\text{err}}(\pi)$ be a function that quantifies the degree of misalignment of a policy $\pi$ with respect to the teacher's expectation, where $J_{\text{err}} = 0$ indicates perfect alignment. In our setting, $J_{\text{err}}(\pi)$ reflects the average number of incorrect intentions proposed by $\pi$, and the negative average score of its intention executions. We define the number of learning episodes $N$ as the smallest integer $n$ such that $J_{\text{err}}(\pi_n) = 0$.

The *long-term communication effort* is defined as the communication effort accumulated across all future learning episodes

$$J(\pi_1) = \mathbb{E}_{\tau_1 \sim \pi_1, \cdots, \tau_N \sim \pi_N} \left[ \sum_{n=1}^{N} C(\tau_n) \right] \tag{1}$$

where $\tau \sim \pi$ denotes generating a learning episode with policy $\pi$, and $\pi_n = \texttt{Improve}(\pi_{n-1}, J)$ with `Improve` being an optimizer (e.g., Adam) that computes a new policy $\pi_n$ such that $J(\pi_n) < J(\pi_{n-1})$. The expectation is taken over all possible sequences of future episodes.

Computing this objective is impractical, so we resort to an approximation scheme. We split $J(\pi_1)$ into two terms, $\mathbb{E}[C(\tau_1)]$ and $\mathbb{E}[C(\tau_2) + \cdots C(\tau_N)]$. The first term, $\mathbb{E}[C(\tau_1)] \triangleq J_{\text{com}}(\pi_1)$, represents the communication effort in the next episode and can be effectively optimized with an RL algorithm. While we cannot directly optimize the second term, we aim to minimize the number of terms in the summation, i.e. minimizing $N$. To do that, we heuristically minimize $J_{\text{err}}(\pi_1)$. The intuition here is that the less misaligned a policy is, the less learning episodes are needed to perfectly align it. This may not always be true but we find the heuristic works sufficiently well in practice. In the end, the current policy $\pi_0$ is updated to a new policy $\pi_1$ that satisfies $J_{\text{com}}(\pi_1) + J_{\text{err}}(\pi_1) < J_{\text{com}}(\pi_0) + J_{\text{err}}(\pi_0)$.

Both terms in the new objective are essential. If trained to minimize only $J_{\text{com}}(\pi_1)$, the learner would quickly resort to taking only a single `[do]` action during an episode. This behavior resembles a sparse-reward RL setting, in which the learner executes the main task and receives a single reward. In this case, because sparsely provided rewards are weak learning signals, the learner would need more learning episodes to master the task, which would result in more long-term communication effort. On the other hand, if the aim is only to reduce $J_{\text{err}}(\pi_1)$, the learner would not be strongly motivated to attempt communication at a higher level of abstraction. It may be content with proposing and executing low-level intentions because those are easy to execute accurately.

**Learning algorithm.** We propose an algorithm that extends Q-learning to optimize for the learner's objective. We define the reward function $r(s, u; i) = -c(s, i, u)$ and the optimal Q-function $Q^\star(s, u; i)$ based on $r$. We approximate this function by $Q_\theta(s, u; i)$ and define the learner's policy as $\pi_\theta(u \mid s, i) = \mathbb{1}\{u = \arg\max_{u'} Q_\theta(s, u'; i)\}$. In each episode, we generate a trajectory using the current policy and store it in a replay buffer. Then we sample a batch of transitions $\{(s_t, i_t, u_t, r_t, s_{t+1}, i_{t+1})\}_{i=1}^{B}$ from the buffer to update the Q-function. To minimize the next-episode communication effort, we apply the standard Q-learning update:

$$\theta_{\text{new}} = \min_\theta \frac{1}{B} \sum_{i=1}^{B} \Big( Q(s_t, u_t; i_t) - (r_t + \gamma \max_u Q_\theta(s_{t+1}, u; i_{t+1})) \Big)^2 \tag{2}$$

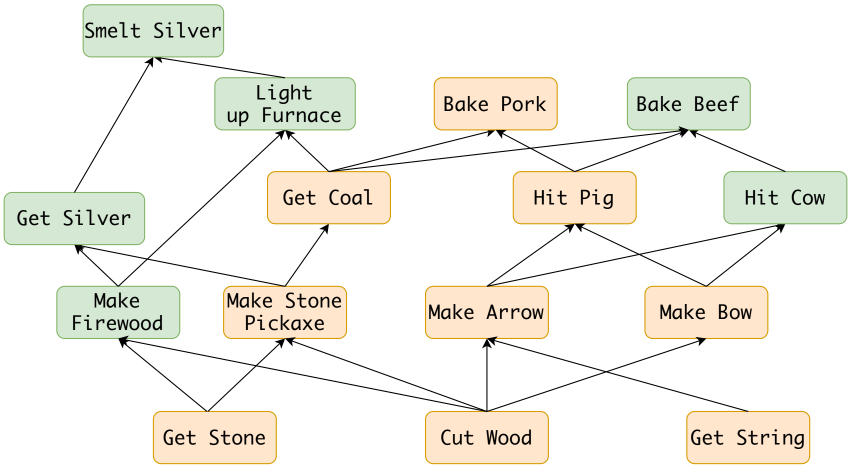
where $\gamma$ is a discount factor. To reduce the task error, we enables the learner to improve by incorporating the teacher's feedback. We consolidate the instructive feedback using a max-margin objective:

$$\theta_{\text{new}} = \min_\theta \frac{1}{B} \sum_{i=1}^{B} \max(0, \lambda + \max_{u \neq \{\texttt{[do]}, u^\star\}} Q(s_t, u; i_t) - Q(s_t, u_t^\star; i_t)) \tag{3}$$

which aims to separate the Q-value of the correct intention $u_t^\star$ from others by a margin of at least $\lambda$. To integrate the evaluative feedback, we implement a weighted self-imitation learning approach, which first computes the objective in Equation 3 with $u_t^\star$ being the primitive actions taken during an execution, and weights this objective by the numerical score provided by the teacher.

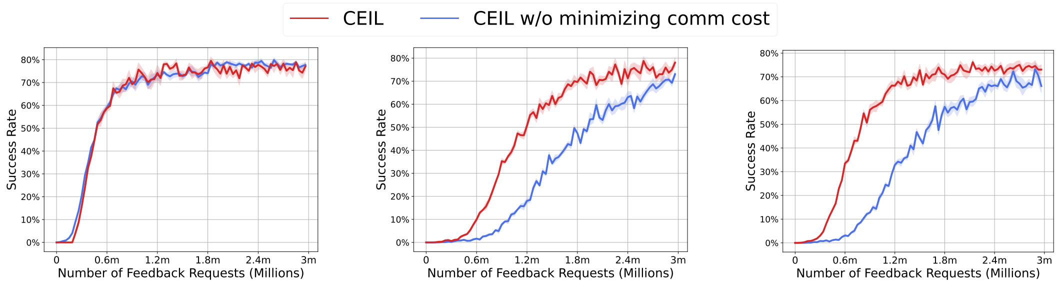
### A.4 Task graph

Below is the complete task graph of the MineCraft environment. Each node represents a task. Each arrow points from a task to its parents, of which it is a subtask. Orange nodes are subtasks of the BakePork task. Green nodes represent subtasks of the two tasks, BakeBeef and SmeltSilver, which the agents learn during the adaptation settings.



### A.5 CEIL without minimizing communication cost

Figure 6 shows the results of CEIL with and without minimizing the next-episode communication effort ($J_{\text{com}}$). The objective term does not noticeably impact the performance with the top-down teacher. However, when learning with the pragmatic teachers, it becomes crucial for enhancing communication efficiency because it effectively guides the learner towards uttering increasingly more abstract intentions.



(a) Top-down (non-pragmatic) teacher

(b) Language-based pragmatic teacher

(c) Performance-based pragmatic teacher

Figure 6: CEIL with and without minimizing next-episode communication effort.

### A.6 Additional implementation details

We implement our baselines and CEIL based on the `stable-baselines3` [Raffin et al., 2021] codebase. We trained all agents with Nvidia 2080Ti GPUs. It took about one and a half days to train CEIL with a 3M feedback-request budget. For the imitation learning baselines, we used an learning rate of $10^{-4}$, and for CEIL, we used $5 \cdot 10^{-5}$. For CEIL, we apply a communication cost of 0.01 per each request for instructive feedback when the student's intention is correct, and 0.05 when it is not. The cost of providing evaluative feedback is 0.2 per request.

Below is the CNN architecture of the state encoder of the learner's policy:

```
state_encoder = nn.Sequential(
    nn.Conv2d(n_input_channels, 16, kernel_size=1, stride=1,
        padding=0),
    nn.ReLU(),
    nn.Conv2d(16, 32, kernel_size=3, stride=2, padding=0),
    nn.ReLU(),
    nn.Conv2d(32, 64, kernel_size=3, stride=1, padding=1),
    nn.ReLU(),
    nn.Conv2d(64, 96, kernel_size=3, stride=1, padding=1),
    nn.ReLU(),
    nn.Conv2d(96, 128, kernel_size=3, stride=1, padding=1),
    nn.ReLU(),
    nn.Conv2d(128, 64, kernel_size=1, stride=1, padding=0),
    nn.ReLU(),
    nn.Flatten(),
    nn.Linear(1024, 256))
)
```

### A.7 Pragmatic teachers

We consider two heuristically pragmatic strategies. With the *language-based* strategy, the teacher aims to speak at the same level of abstraction as the learner. We define the level of abstraction of an intention as the optimal number of actions required to complete the corresponding task. Let $T^{\star}_{u}$ be the level of abstraction of intention $u$. The teacher samples an intention to return according to the probability distribution $P(u) \propto |T^{\star}_{u} - T^{\star}_{\hat{u}}|^{-1}$ where $T^{\star}_{\hat{u}}$ is the level of abstraction of the learner's proposed intention. With the *performance-based* strategy, the teacher records the moving success rate $\rho_u$ of the learner in executing each intention $u$. The returned intention is sampled according to $P(u) \propto \rho_u$. Thus, the better the learner is at executing an intention, the more likely the teacher is to refer to that intention when instructing it.