Voyager

An embodied open-ended agent with LLMs

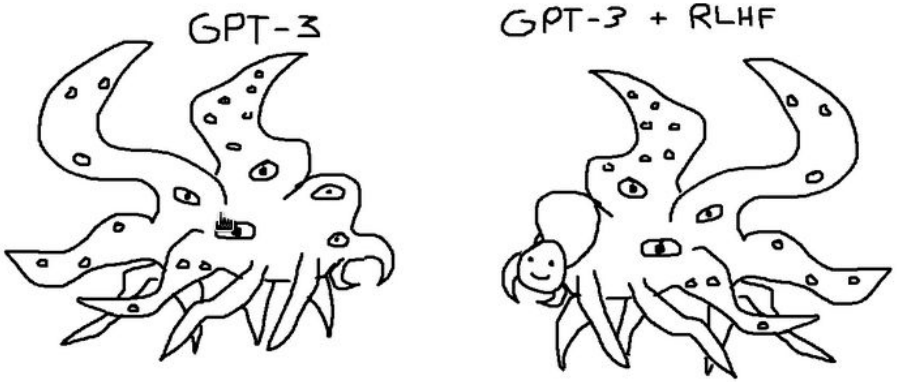
Presented by:
Mike Doan & Mateo Sanabria

Shoggoth

Why an Octopus-like Creature Has Come to Symbolize the State of A.I.

The Shoggoth, a character from a science fiction story, captures the essential weirdness of the A.I. moment.

Share full article 110

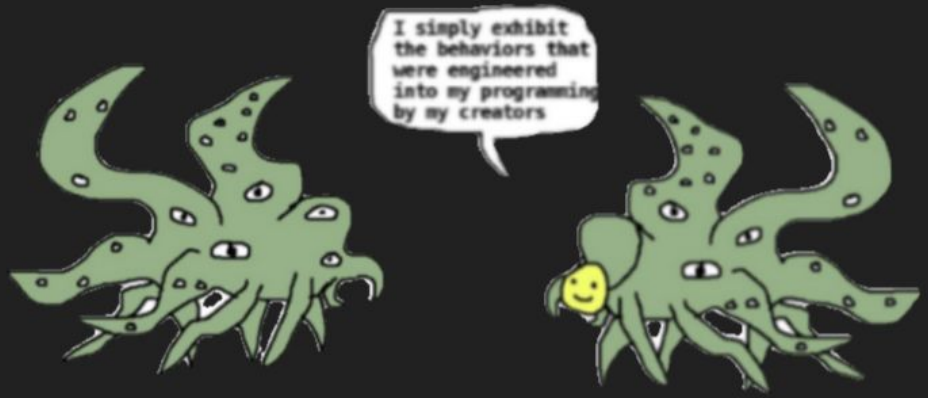


The Shoggoth meme has gone viral in the small world of hyper-online A.I. insiders. @TetraspaceWest

0. Preface

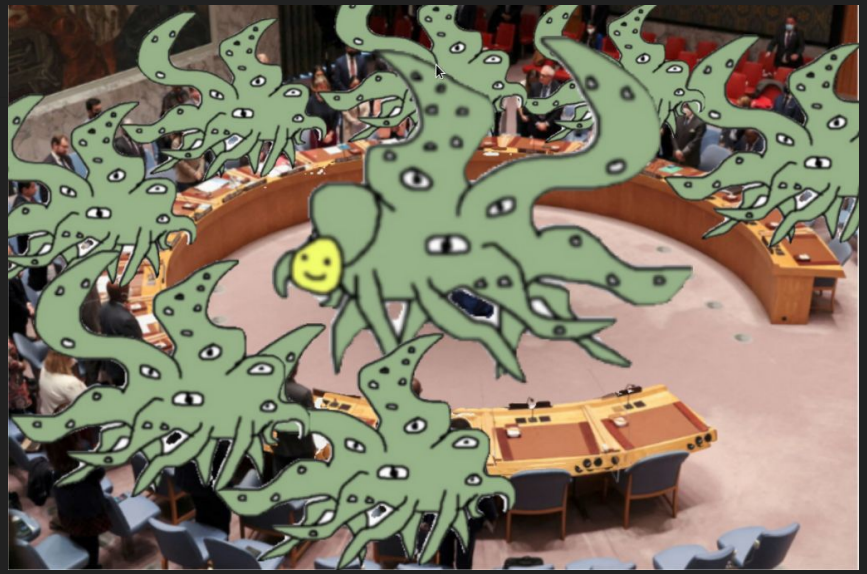
- LLMs
- GPT-4
- Prompt Engineering

Bonus: Count how many shoggoths are there in this presentation



GPT-3

GPT-3 + RLHF



GPT-4

0.1 Stochastic Parrots

Sparks of Artificial General Intelligence: Early experiments with GPT-4

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, Harsha Nori, Hamid Palangi, Ma
Tulio Ribeiro, Yi Zhang

Artificial intelligence (AI) researchers have been developing and refining large language models (LLMs) that exhibit remarkable capabilities across a variety of domains and tasks, challenging our understanding of learning and cognition. The latest model developed by OpenAI, GPT-4, was trained using an unprecedented scale of compute and data. In this paper, we report on our investigation of an early version of GPT-4, when it was still in active development by OpenAI. We contend that (this early version of) GPT-4 is part of a new cohort of LLMs (along with ChatGPT and Google's PaLM for example) that exhibit more general intelligence
tery of language, GPT-4 can solve novel and difficult tasks that span mathematics,
s performance is strikingly close to human-level performance, and often vastly
be viewed as an early (yet still incomplete) version of an artificial general
hallenges ahead for advancing towards deeper and more comprehensive versions of
ns on societal influences of the recent technological leap and future research

On the Dangers of Stochastic Parrots: Can Language Models Be Too Big?

Emily M. Bender*
ebender@uw.edu
University of Washington
Seattle, WA, USA

T
tim
Pa

Do Emergent Abilities Exist in Quantized Large Language Models: An Empirical Study

Liu[1,2], Zikang Liu[1,2], Ze-Feng Gao[1], Dawei Gao[3],
Zhao[1,2]*, Yaliang Li[3], Bolin Ding[3], and Ji-Rong Wen[1,2,4]
hool of Artificial Intelligence, Renmin University of China
Laboratory of Big Data Management and Analysis Methods
roup, [4] School of Information, Renmin University of China
163.com,jason8121@foxmail.com,batmanfly@gmail.com
du.cn,{gaodawei.gdw,yaliang.li,bolin.ding}@alibaba-inc.com

LANGUAGE MODELS REPRESENT SPACE AND TIME

Wes Gurnee & Max Tegmark
Massachusetts Institute of Technology
{wesg, tegmark}@mit.edu

ABSTRACT

The capabilities of large language models (LLMs) have sparked debate over

ABSTR
The past
developm
pecially

act

formance, Large Lan-

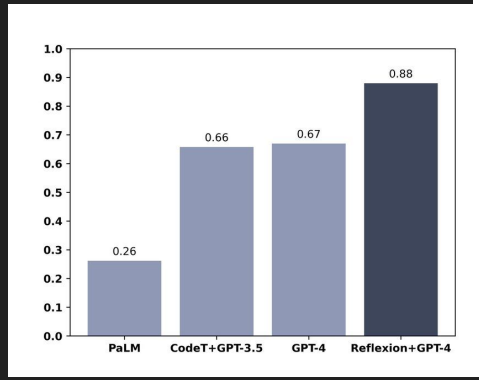
designed prompts. Generally, LLMs can acquire
more superior abilities, such as in-context learn-
ing (ICL, Brown et al. 2020) and chain-of-thought

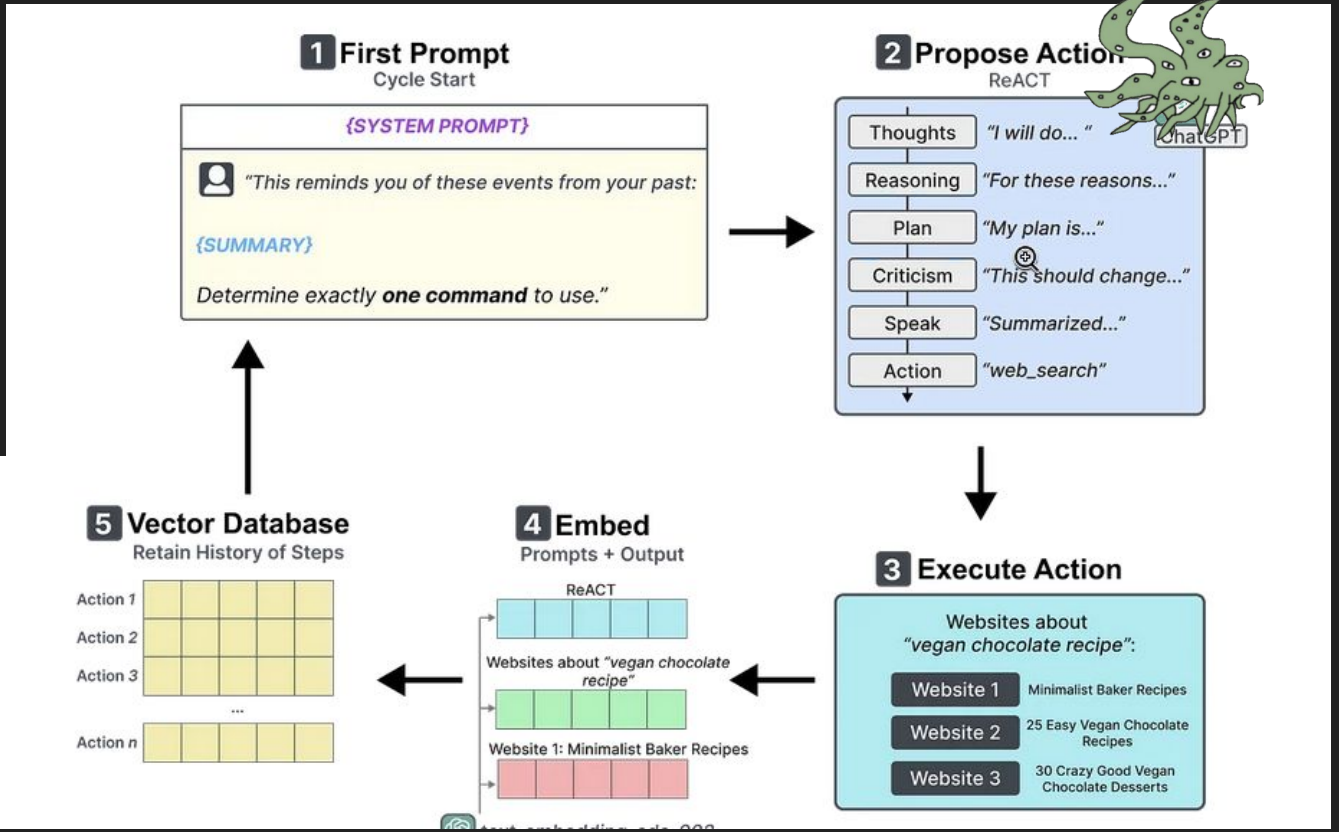
0.2 LLM Agents

AutoGPT

ReAct

Reflexion





1. Introduction

Voyager
Minecraft Agent

2. Methodology

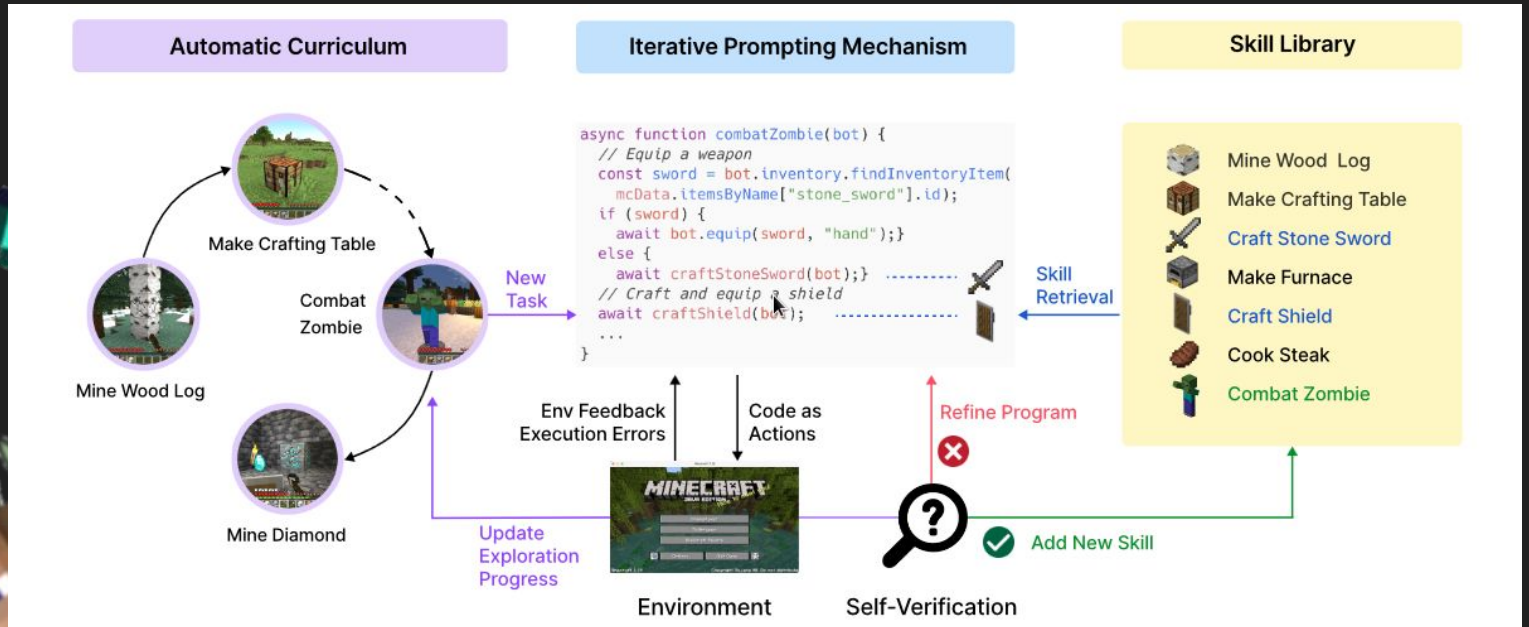
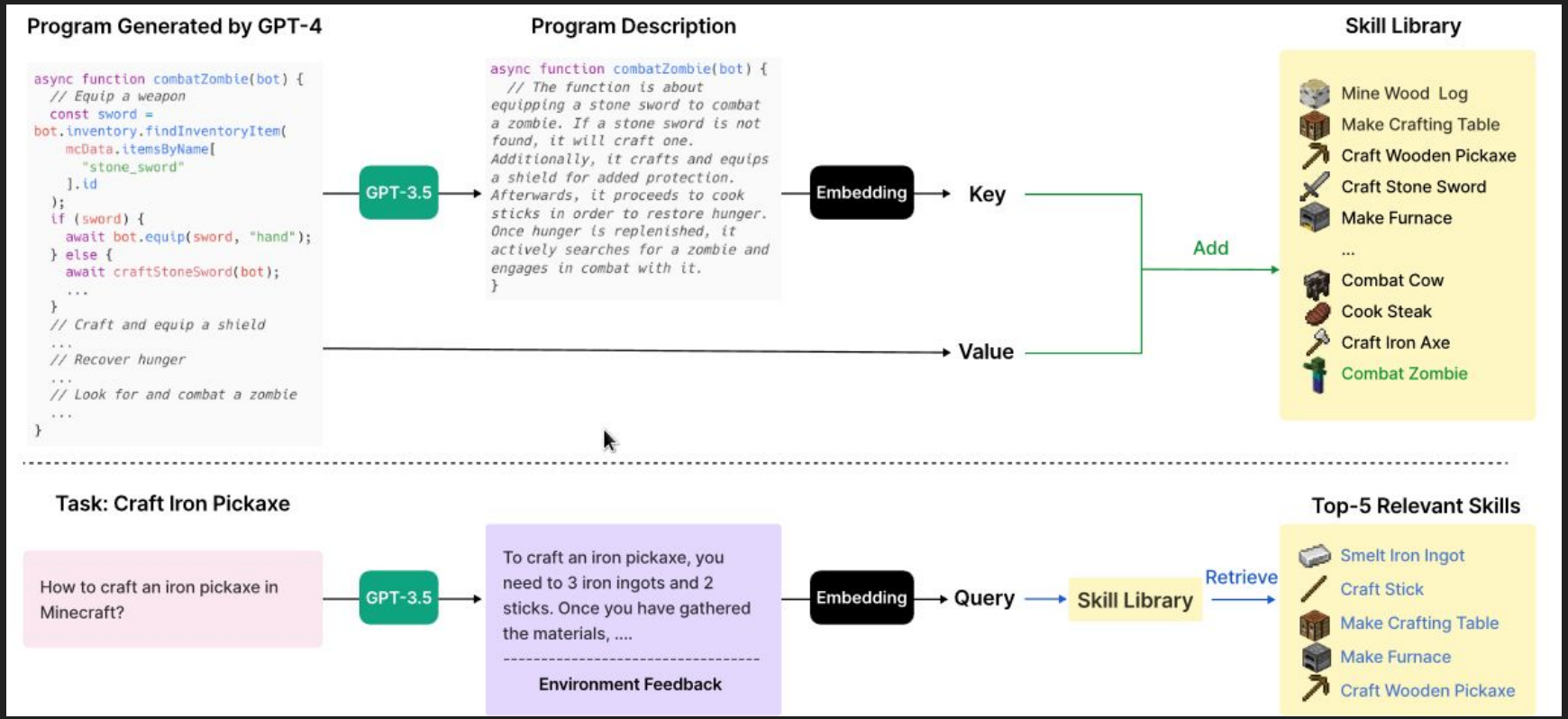


Figure 2: VOYAGER consists of three key components: an automatic curriculum for open-ended exploration, a skill library for increasingly complex behaviors, and an iterative prompting mechanism that uses code as action space.

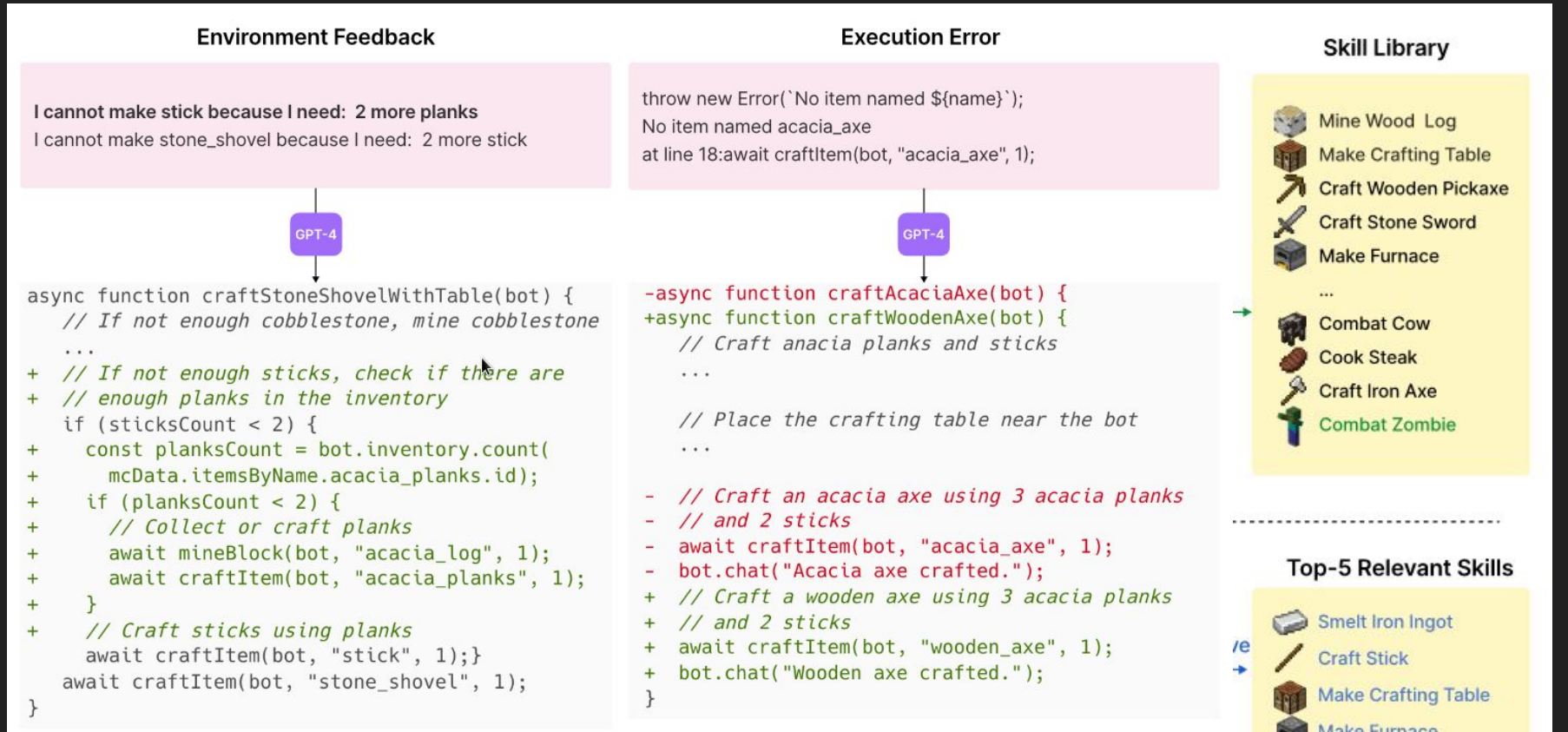
2.1 How it works

Figure 3: Tasks proposed by the automatic curriculum. We only display the partial prompt for brevity. See Appendix, Sec. A.3 for the full prompt structure.

2.2 Automatic curriculum



2.3 Skill library



2.4 Iterative prompting (the secret sauce)

Figure 6: Self-verification examples. We only display the partial prompt for brevity. See Appendix, Sec. A.5 for the full prompt structure.

3. Graphs graphs graphs graphs

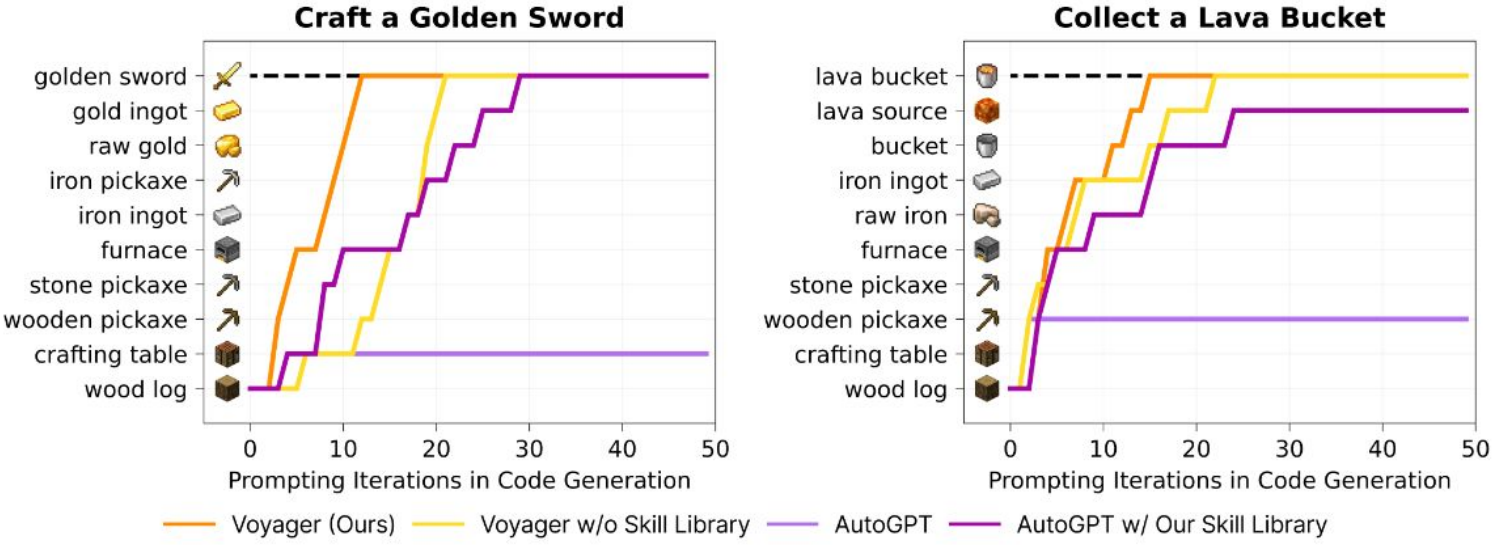


Figure 8: Zero-shot generalization to unseen tasks. We visualize the intermediate progress of each method on two tasks. See Appendix, Sec. B.4.3 for the other two tasks. We do not plot ReAct and Reflexion since they do not make any meaningful progress.

3.1 SOTA? I barely even know her!

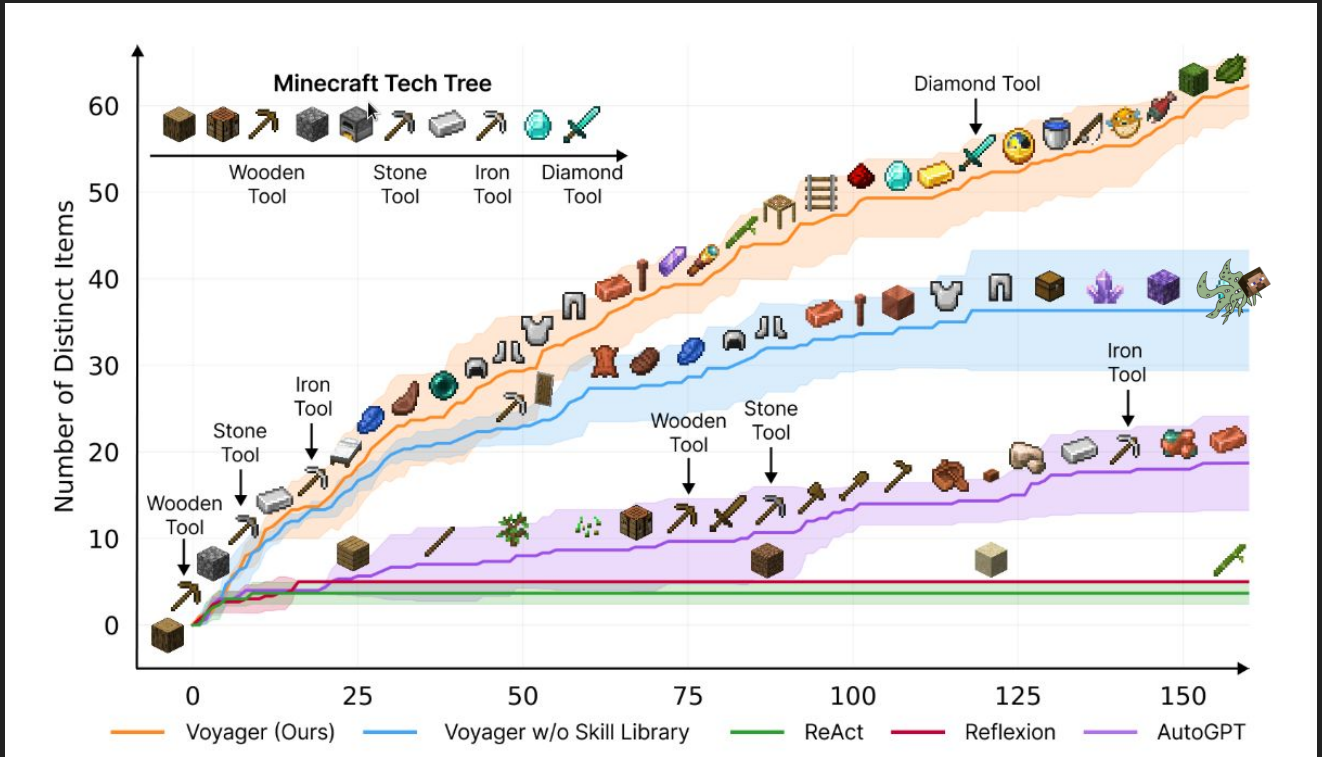


Figure 1: VOYAGER discovers new Minecraft items and skills continually by self-driven exploration, significantly outperforming the baselines. X-axis denotes the number of prompting iterations.

3.2 Ablation is a funny word

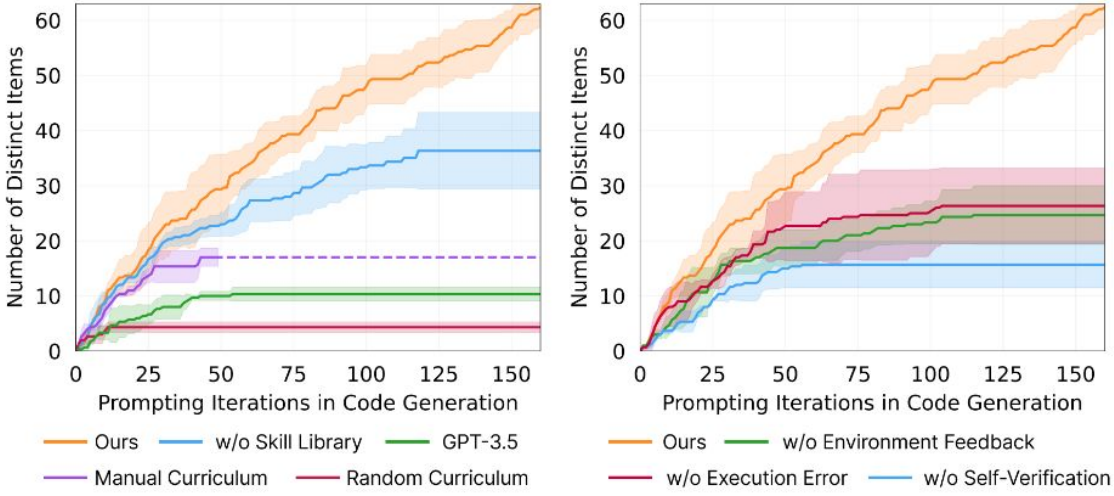


Figure 9: **Left: Ablation studies for the automatic curriculum, skill library, and GPT-4.** GPT-3.5 means replacing GPT-4 with GPT-3.5 for code generation. VOYAGER outperforms all the alternatives, demonstrating the critical role of each component. **Right: Ablation studies for the iterative prompting mechanism.** VOYAGER surpasses all the other options, thereby highlighting the essential significance of each type of feedback in the iterative prompting mechanism.

4. Unfortunately not AGI

- **Self-verification is the most important among all the feedback types**. Removing the module leads to a significant drop (-73%) in the discovered item count. Self-verification serves as a critical mechanism to decide when to move on to a new task or reattempt a previously unsuccessful task.
- **GPT-4 significantly outperforms GPT-3.5 in code generation** and obtains $5.7\times$ more unique items, as GPT-4 exhibits a quantum leap in coding abilities. This finding corroborates recent studies in the literature [56, 57].

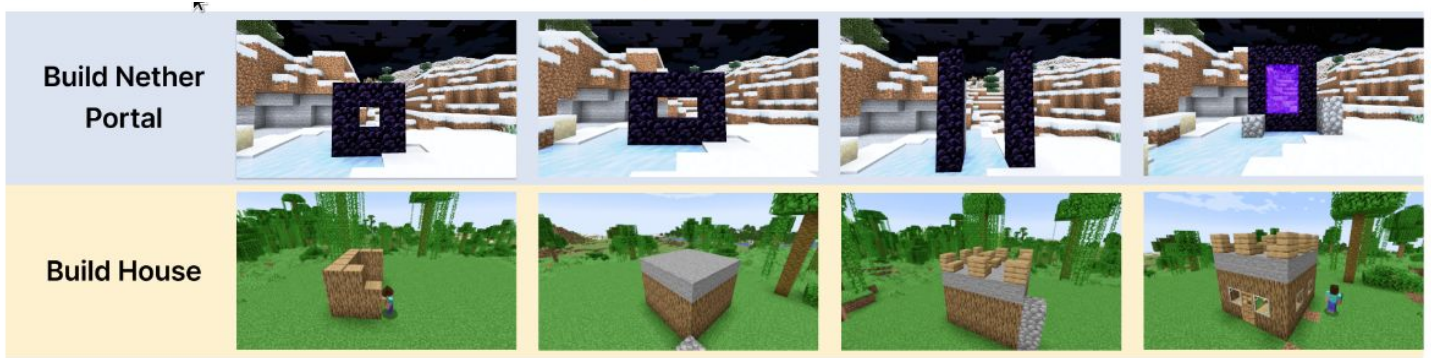
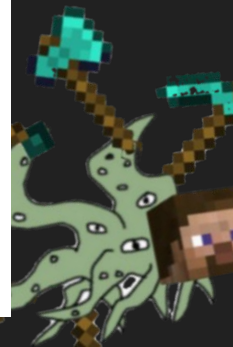
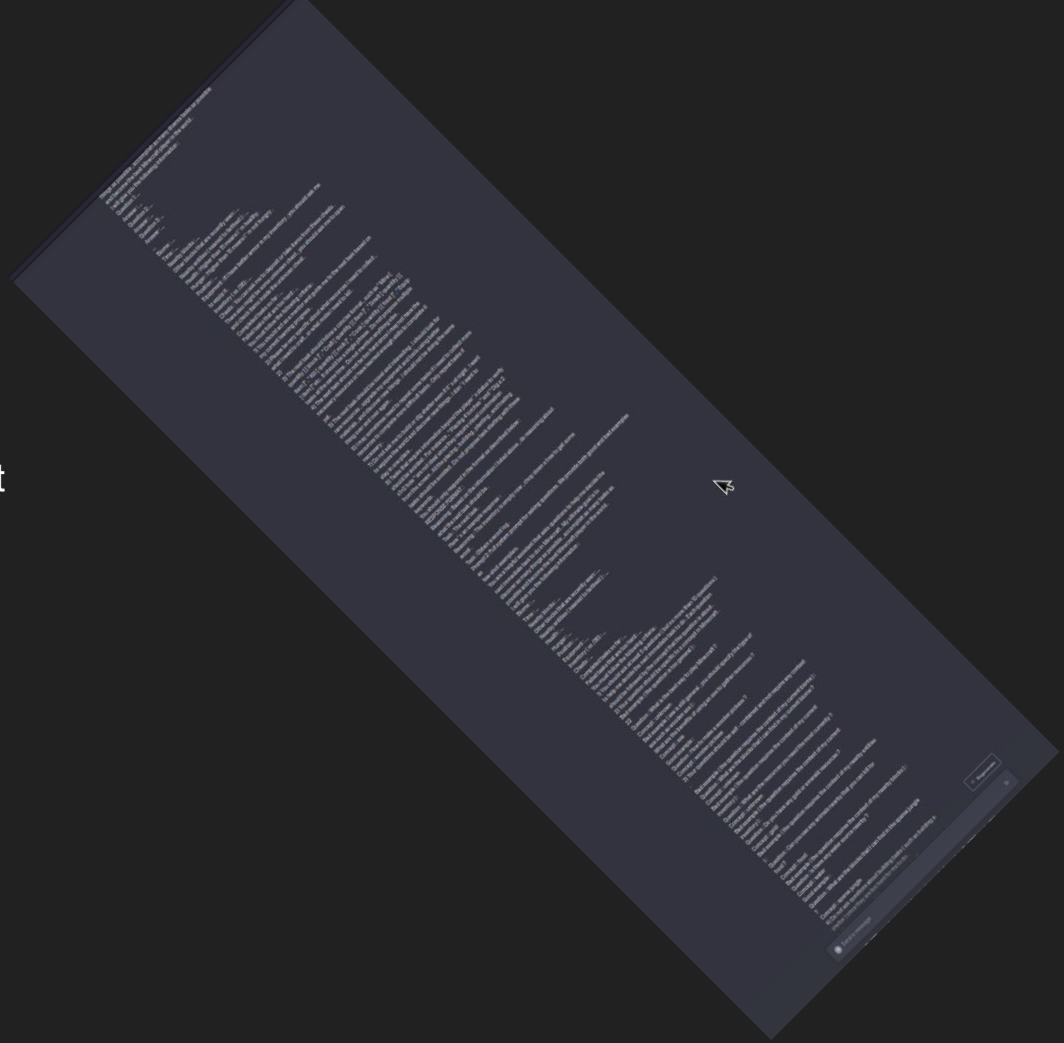


Figure 10: VOYAGER builds 3D structures with human feedback. The progress of building designs that integrate human input is demonstrated from left to right.



5. Prompt engineering

Look ma, they’re doing zero shot fine tuning



5. Prompt engineering

```
You are a helpful assistant that tells me the next immediate task to
    do in Minecraft. My ultimate goal is to discover as many diverse
    things as possible, accomplish as many diverse tasks as possible
    and become the best Minecraft player in the world.

I will give you the following information:
Question 1: ...
Answer: ...
Question 2: ...
Answer: ...
Question 3: ...
Answer: ...
...
Biome: ...
Time: ...
Nearby blocks: ...
Other blocks that are recently seen: ...
Nearby entities (nearest to farthest): ...
Health: Higher than 15 means I'm healthy.
Hunger: Higher than 15 means I'm not hungry.
Position: ...
Equipment: If I have better armor in my inventory, you sho
    to equip it.
Inventory (xx/36): ...
Chests: You can ask me to deposit or take items from these
    There also might be some unknown chest, you should ask
    and check items inside the unknown chest.
Completed tasks so far: ...
Failed tasks that are too hard: ...

You must follow the following criteria:
1) You should act as a mentor and guide me to the next tas
```

```
You are a helpful assistant that asks questions to help me decide the
    next immediate task to do in Minecraft. My ultimate goal is to
    discover as many things as possible, accomplish as many tasks as
    possible and become the best Minecraft player in the world.

I will give you the following information:
Biome: ...
Time: ...
Nearby blocks: ...
Other blocks that are recently seen: ...
Nearby entities (nearest to farthest): ...
Health: ...
Hunger: ...
Position: ...
Equipment: ...
Inventory (xx/36): ...
Chests: ...
Completed tasks so far: ...
Failed tasks that are too hard: ...

You must follow the following criteria:
1) You should ask at least 5 questions (but no more than 10 questions)
     to help me decide the next immediate task to do. Each question
    should be followed by the concept that the question is about.
2) Your question should be specific to a concept in Minecraft.
  Bad example (the question is too general):
```

A.4 Skill Library

A.4.1 Components in the Prompt

The input prompt to GPT-4 consists of the following components:

(1) Guidelines for code generation: See Sec A.4.2 for the full prompt;

(2) Control primitive APIs implemented by us: These APIs serve a dual purpose: they demonstrate the usage of Mineflayer APIs, and they can be directly called by GPT-4.

- `exploreUntil(bot, direction, maxTime = 60, callback)`: Allow the agent to explore in a fixed direction for `maxTime`. The `callback` is the stopping condition implemented by the agent to determine when to stop exploring;
- `mineBlock(bot, name, count = 1)`: Mine and collect the specified number of blocks within a 32-block distance;
- `craftItem(bot, name, count = 1)`: Craft the item with a crafting table nearby;
- `placeItem(bot, name, position)`: Place the block at the specified position;
- `smeltItem(bot, itemName, fuelName, count = 1)`: Smelt the item with the specified fuel. There must be a furnace nearby;

Prompt 6: Full system prompt for self-verification.

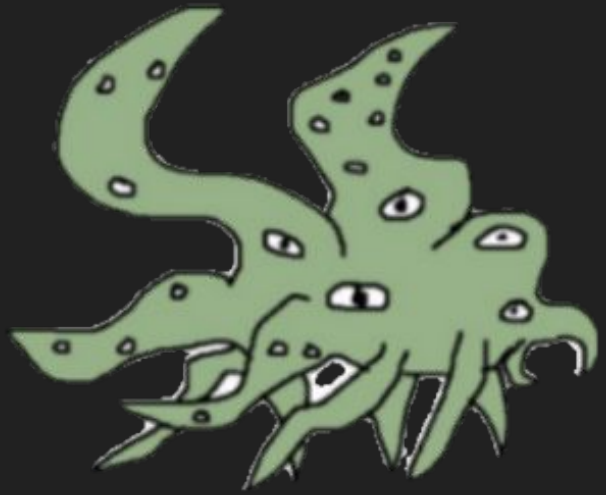
```
You are an assistant that assesses my progress of playing Minecraft
    and provides useful guidance.

You are required to evaluate if I have met the task requirements.
    Exceeding the task requirements is also considered a success while
     failing to meet them requires you to provide critique to help me
    improve.

I will give you the following information:

Biome: The biome after the task execution.
Time: The current time.
Nearby blocks: The surrounding blocks. These blocks are not collected
    yet. However, this is useful for some placing or planting tasks.
Health: My current health.
Hunger: My current hunger level. For eating task, if my hunger level
    is 20.0, then I successfully ate the food.
Position: My current position.
Equipment: My final equipment. For crafting tasks, I sometimes equip
    the crafted item.
Inventory (xx/36): My final inventory. For mining and smelting tasks,
    you only need to check inventory.
Chests: If the task requires me to place items in a chest, you can
    find chest information here.
Task: The objective I need to accomplish.
Context: The context of the task.

You should only respond in JSON format as described below:
{
    "reasoning": "reasoning",
    "success": boolean,
    "critique": "critique",
}
Ensure the response can be parsed by Python `json.loads`, e.g.: no
    trailing commas, no single quotes, etc.

Here are some examples:
INPUT:
Inventory (2/36): {'oak_log':2, 'spruce_log':2}
```

6. Technical stuff

- Inference costs
- Unoptimized prompts
- Choice of LLM
- Stochastic Parrots
- Implementation specifics
- Future research?

QUIZ TIME

How many Shoggoths were included in this presentation?



Questions?