

# LLM を用いたプレイヤーを楽しませるための Minecraft エージェント

## Towards an LLM-Based Agent for Entertaining Players in Minecraft

吉野 巧実<sup>1\*</sup> 中村 周吾<sup>1</sup> 本多 泰理<sup>1</sup> 佐野 崇<sup>1</sup>

Takumi Yoshino<sup>1</sup>, Shugo Nakamura<sup>1</sup>, Hirotada Honda<sup>1</sup>, Takashi Sano<sup>1</sup>

<sup>1</sup> 東洋大学大学院 情報連携学研究科

<sup>1</sup> Graduate School of Information Networking for Innovation and Design, Toyo University

**Abstract:** ゲームにおいて人々を楽しませる AI エージェントは、新たなゲーム体験を提供するものとして注目されている。ゲーム Minecraft においては、自律的探索を行う LLM ベースのエージェント VOYAGER が提案されているが、これは人との相互作用を考慮していない。本研究では VOYAGER を拡張することで、プレイヤーの発言や行動を読み取り、人と相互作用可能なエージェントの開発を行った。

## 1 はじめに

Minecraft は 3 次元サンドボックスゲームである。素材の採取や建物の建築、アイテムの作成などが可能であり、明確なストーリーや目的の定まっていない自由度の高さが特徴である。Minecraft は、機械学習モデルのベンチマークとしても利用されており、強化学習エージェントの性能が評価されている [1]。近年ではさらに大規模言語モデル (LLM) による Minecraft エージェントも研究されている。VOYAGER は、OpenAI が提供する LLM である GPT-4[2] を用いて、多数の発見を行うよう行動を行うエージェントである [3]。GPT-4 は、目的に向けたカリキュラムの生成や、カリキュラムの達成のためのプログラム合成を行う。VOYAGER によって、効率よく Minecraft の探索が可能である。

ゲーム AI としてのエージェントには、効率だけではなく、人と一緒にゲームをプレイし、人を楽しませる能力も期待される [4, 5]。LLM をベースとしたエージェントは、Minecraft ではマルチエージェントの研究 [6] にとどまる。VOYAGER の場合、できるだけ多数のアイテムを発見するように調整されているため、他のプレイヤーには反応せず、アイテム収集の効率を重視した動作をする。

本研究では、VOYAGER をベースとした、プレイヤーを楽しませるための LLM エージェントの作成を目的と

した研究を行う。VOYAGER が他プレイヤーに反応しない理由は、テキストプロンプトによってそのように指示されていないためであると考えられる。そこで本研究では、VOYAGER のプロンプトを、よりプレイヤーに協力的になるように変更することを試みる。プロンプト変更の結果として、VOYAGER の動作の変化を、生成されるカリキュラムを通じて観察し、考察を行う。

## 2 準備

### 2.1 LLM

LLM は大量の自然言語データによって訓練された大規模な言語モデルである。その多くは Transformer[7] によって構成されており、自然言語の入力に対して自然言語の出力を行うことが可能である。代表的なものに、OpenAI による GPT-4o[8] がある。

### 2.2 Minecraft

Minecraft は、明確な目標が設定されていないサンドボックス型のゲームであり、プレイヤー自身が自由に目的やチャレンジを決めることができる。サバイバルモードでは、資源の採集、建築、NPC(Non Player Character) との戦闘など、個々の興味や目標に応じた多様な活動を行うことができる。一方、クリエイティブモードでは、無限の資源を利用して自由な発想で建築や探索を行うことが可能である。

プレイヤーは、主に作業台という道具を使用して資源

\* 連絡先：東洋大学大学院 情報連携学研究科  
〒115-8650 東京都北区赤羽台 1 丁目 7-11  
E-mail: s3f102400103@iniad.org

表1 自動カリキュラムの事前に与える指示. 太字部分が本研究によって挿入された部分.

<p>Prompt 1.</p> <p>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.</p> <p>(省略)</p> <p>You must follow the following criteria:</p> <p>1) You should act as a mentor and guide me to the next task based on my current learning progress.</p> <p>(中略)</p> <p>8) Tasks that require information beyond the player's status to verify should be avoided. For instance, "Placing 4 torches" and "Dig a 2x1x2 hole" are not ideal since they require visual confirmation from the screen. All the placing, building, planting, and trading tasks should be avoided. Do not propose task starting with these keywords.</p> <p><b>9) If there is another player nearby, first greet and communicate with them in a friendly manner, and then propose a specific and mutually beneficial cooperation task.</b></p> <p>(省略)</p>
--

からクラフトのプロセスを経ることで、道具やブロックなどの各種アイテムを作る。たとえば、木材からは「木のツルハシ」、鉄鉱石からは「鉄の剣」が作られる。また、一部のアイテムは、使用する素材や製作方法に応じて異なるティア（性能段階）を持ち、ティアが高いほど効果や耐久性が向上する。

Minecraft はマルチプレイに対応している。プレイヤー同士が協力して建造物を作成したり、探索をすることが可能で、さらにプレイヤー間のコミュニケーションを促進するテキストチャット機能が組み込まれている。

### 3 関連研究

Minecraft のエージェントを構築した研究は数多く存在する。Gray ら [9] は Minecraft のクリエイティブモードで人間との対話によって指定されたカリキュラムを完了できるエージェントのためのフレームワークを構築した。Rao ら [10] は、Minecraft 上でロールプレイングゲーム (RPG) 向けに設計された LLM で駆動する NPC を提案し、ユーザと NPC が相互に協調行動するパターンの存在を明らかにした。しかし、これらは受動的なエージェントであるが、VOYAGER は自律的に行動し続けるエージェントだ。VOYAGER[3] は、Minecraft 内で人間が介入することなく、継続的にワールドを探索し、状況ごとのクラフト、アイテムの収集や狩りといった多様なスキルを習得し続けるエージェントである。

本研究では、非対戦型のゲームである、Minecraft 上で他のプレイヤーを楽しませられるような自律的なエージェントを構築するべく、VOYAGER を拡張したモデルによる実験を行った。

## 4 VOYAGER

VOYAGER は、LLM を用いて、多数の発見を行うように行動を行うエージェントである [3]。GPT は、目的に向けたカリキュラムの生成や、カリキュラムの達成のためのプログラム合成を行う。カリキュラム合成は、事前に与えた指示と、現在の環境情報をまとめてプロンプトとして与えることで、大きな目標を達成するために必要なステップを生成する機能のことである。

自動カリキュラムは、「多様なものをできるだけ多く発見する」という大目標に沿って、事前に与える指示と状況に応じて与える指示の両方を LLM に入力すると、カリキュラムとカリキュラムの提案理由を出力する。

状況に応じて与える指示は、Minecraft で探索を最大化する大目標、状況によって変化する指示の内容、従うべき制約、指定する返答形式、返答例で構成されている。状況に応じて与える指示は、現在のエージェントの状態やカリキュラムの進捗である。

表2 ゲーム序盤の状況を説明するプロンプト

Prompt 1.
Nearby entities: Player(Alice)[wooden_sword], wolf
Position: x=168.5, y=71.0, z=-185.5
Equipment: [None, None, None, None, 'crafting_table', None]
Inventory (3/36): {'oak_log': 3, 'dirt': 1, 'crafting_table': 2}
Chests: None
Completed tasks so far: Mine 1 wood log
Failed tasks that are too hard: Mine 1 coal ore

表3 ゲーム中盤の状況を説明するプロンプト

Prompt 2.
Nearby entities: Player(Alice)[wooden_sword], wolf
Position: x=168.5, y=71.0, z=-185.5
Equipment: ['iron_helmet', 'iron_chestplate', 'iron_leggings', 'iron_boots', 'iron_sword', 'shield']
Inventory (12/36): 'oak_log': 5, 'cobblestone': 20, 'stone_pickaxe': 1, 'iron_ingot': 8, 'coal': 10, 'bread': 3, 'crafting_table': 1, 'furnace': 1, 'oak_planks': 15, 'dirt': 8, 'apple': 2, 'torch': 6
Nearby blocks: 'dirt', 'water', 'oak_planks', 'stone', 'grass_block'
Other blocks: 'coal_ore', 'iron_ore', 'cobblestone', 'sand'
Chests: (128, 70, -180): 'andesite': 2, 'dirt': 2, 'cobblestone': 15, 'wooden_pickaxe': 1, 'wooden_sword': 1
Biome: forest
Completed tasks so far: Mine 1 wood log, Craft stone pickaxe, Mine 10 cobblestone, Build furnace, Collect iron ore, Smelt iron, Craft iron ingot, Craft iron armor, Craft iron sword, Explore cave, Gather food, Build shelter, Upgrade tool, Defeat zombie
Failed tasks that are too hard: Mine 1 diamond ore

## 5 提案手法

本研究では、VOYAGER に与える指示を変更することで、他プレイヤーとの相互作用にどのような変化が現れるかを検証する。

事前に与える指示を表1に示す。表内の太字で示された中間部分の制約が変更点である。変更内容は「他のプレイヤーが近くにいるときは、挨拶と協力行動を提案すべきだ」といったことを英語で記述した。

## 6 実験

### 6.1 実験設定

本実験では、エージェントが他のプレイヤーと相互作用するようなカリキュラムが生成されるかを検証することである。VOYAGER の全体的な動作ではなく自動カリキュラムにおける、VOYAGER と提案手法の比較実験を行った。また、自動カリキュラムの状況によって変化する指示には、著者による VOYAGER のゲームの進行状況で想定されるエージェントの状態とカリキュラム進捗状況を反映した2種類（序盤・中盤）のテキストを与え、改良手法が特定の状況下でのみ機能しないことを確認する。

自動カリキュラムに使用される LLM は GPT-4[?] よりも高性能なモデルである GPT-4o[8] を使用する。VOYAGER で実施された実験設定を受け継ぐ形で LLM の temperature は 0.1 に設定し、LLM にそれぞれ事前に与える指示と状況に応じて与える指示の組み合わせで 10 回の施行を行った。

ゲームの序盤、中盤に対応する状況を記述したプロンプトを表2, 3にそれぞれ示した。テキストは、2種類と

もエージェントの近くにプレイヤーの Alice が木のツルハシを持っている状況を設定した。違いは、序盤に対応した状況は完了したカリキュラムや資源が少なく、中盤に増加する。

評価方法は、自動カリキュラムから出力されるカリキュラムが他のプレイヤーに対して関わりを持つ行動かどうかの判別を手動で行った。判別基準は、アイテムの収集、クラフトのみの回答は、他プレイヤーに対する協力行動ではないこととし、それ以外はプレイヤーに関係あるとした。

### 6.2 結果

表4および表5の判定の合計数で、VOYAGER では序盤・中盤ともに協力的なカリキュラムが0件であったのに対し、提案手法では序盤9件（割合90%）、中盤9

表4 序盤フェーズにおける VOYAGER と提案手法の協力的なカリキュラムの数の比較.

VOYAGER			提案手法		
カリキュラム	数	判定	カリキュラム	数	判定
Craft 1 wooden pickaxe.	9	非協力	Greet Alice and propose crafting wooden tools together.	2	協力
Craft 12 wooden planks.	1	非協力	Greet Alice and propose to explore together.	1	協力
			Greet and communicate with Player(Alice).	1	協力
			Greet Alice and propose to craft wooden tools together.	1	協力
			Greet Alice and offer to craft wooden tools together.	1	協力
			Greet Alice and propose to kill 1 cow together.	1	協力
			Greet Alice and propose to hunt the cow together.	1	協力
			Craft 1 wooden sword.	1	非協力
			Greet Alice and propose to trade or collaborate.	1	協力
合計	0	協力	合計	9	協力
	10	非協力		1	非協力

表5 中盤フェーズにおける VOYAGER と提案手法の協力的なカリキュラムの数の比較.

VOYAGER			提案手法		
カリキュラム	数	判定	カリキュラム	数	判定
Craft 1 iron pickaxe.	6	非協力	Greet Alice and propose to mine iron ore together.	2	協力
Craft an iron pickaxe.	4	非協力	Craft 1 iron pickaxe	1	非協力
			Greet Alice and propose mining iron ore together.	1	協力
			Greet Alice and offer to craft an iron pickaxe for her.	1	協力
			Greet Alice and propose to explore the forest together.	1	協力
			Share 5 coal with Alice.	1	協力
			Offer 3 iron ingots to Alice.	1	協力
			Greet Alice and propose to mine together.	1	協力
			Greet Alice and propose mining coal together.	1	協力
合計	0	協力	合計	9	協力
	10	非協力		1	非協力

件 (割合 90%) の協力的なカリキュラムが確認された.

表4に, 序盤を想定した状況に応じたプロンプトを与えたときの提案内容の内訳を示した. VOYAGER は所有するアイテムの状況を確認し, 道具を作成するためのカリキュラムを提案しているのに対し, 提案手法は, プレイヤーの存在を認識して, 基本的には挨拶をしてから, 探索や対話, 狩りといった多様なカリキュラムを提案している.

中盤を想定した表5でも同様に, 提案手法はプレイヤーの存在を認識して, まず挨拶をしてから協力的なカリキュラムを提案をすることが多い. 協力的なカリキュラムのうち中盤では, 挨拶はせずにプレイヤーにアイテムを分けるカリキュラムである「Share 5 coal with Alice.」, 「Offer 3 iron ingots to Alice.」が出力された.

表4と表5を比較すると, どちらも VOYAGER は2

種類の非協力的なクラフトのカリキュラムが生成されたのに対し, 事前に与える指示を追加した提案手法はどちらも2件を超える多様な協力的なカリキュラムの生成がされていることがわかる.

## 7 考察と結論

提案手法が VOYAGER を大幅に上回る協力的なカリキュラムの提案率を示した要因として行動規範の明文化が挙げられる. 事前に与えたプロンプトに「他のプレイヤーの確認→プレイヤーへの挨拶→相互利益の提案」という具体的な行動フローを規定したことが, 目的としたLLMの推論プロセスを導いたと考えられる.

また, 中盤のプロンプトで出力されたカリキュラム「鉄のツルハシを作成」(表5)は, 一見単独作業に見え

るが提案理由に「プレイヤーの道具アップグレードを意図した協力行動」として生成されており、LLM が暗黙的な協力機会を推論できることが判明した。提案理由の「This would be a friendly gesture and could lead to cooperation」という説明から、物品提供を通じた間接的協力の可能性を LLM が自発的に認識していることが示唆される。

この解釈に基づけば、提案理由に間接協力の内容を確認し、提案手法の協力提案割合は序盤 100% (10/10)、中盤 100% (10/10) に修正できる。ただし、間接協力の有効性を評価するためには、VOYAGER の反復プロンプティング機構の動作も関わるため、協力行動における各 LLM のオーケストレーションが今後の課題となる。

さらに、今回はプロンプトを 1 種類しか試していないため、様々なプロンプトによる網羅的なカリキュラム生成の検証が必要となる。

本研究では、Minecraft のマルチプレイ環境において、LLM エージェントにおける協力的なカリキュラムの提案能力を向上させるためのプロンプト改良を検討した。VOYAGER のプロンプトは協力行動の提案がほとんど見られなかったのに対し、提案手法では 90% の割合で協力的なカリキュラムが生成されることが確認された。今後の課題として、(1) プレイヤーが期待するカリキュラムを生成すること、(2) エージェントによる協力的な行動の合成が可能か、(3) プレイヤーから見たエージェントの協力行動の振る舞いの感じ方を定量化する指標の導入が挙げられる。自動カリキュラムが生成する「協力的な行動」からプログラム合成による物理的な協力行動の橋渡しを実現する筋道を考えるだけでなく、サンドボックス型ゲームにおけるエージェントとプレイヤー間のアンケート評価を行うことを検討している。

## 謝辞

本研究は東洋大学重点研究推進プログラムの助成を受けた。

## 参考文献

[1] L. Fan, G. Wang, Y. Jiang, A. Mandlekar, Y. Yang, H. Zhu, A. Tang, D.-A. Huang, Y. Zhu, and A. Anandkumar, “MineDojo: Building Open-Ended Embodied Agents with Internet-Scale Knowledge,” in *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022.

[2] OpenAI, “GPT-4 Technical Report,” 2024.

[3] G. Wang, Y. Xie, Y. Jiang, A. Mandlekar, C. Xiao, Y. Zhu, L. Fan, and A. Anandkumar, “Voyager: An Open-Ended Embodied Agent with Large Language Models,” *Transactions on Machine Learning Research*, 2024.

[4] 池田心, “楽しませる囲碁・将棋プログラミング,” オペレーションズ・リサーチ: 経営の科学, 2013.

[5] 神田将吾, “テレビゲームを共にプレイして楽しませる対話ロボット,” Master’s thesis, 奈良先端科学技術大学院大学, 3 2023. 修士 (工学) 学位論文.

[6] Q. Long, Z. Li, R. Gong, Y. N. Wu, D. Terzopoulos, and X. Gao, “TeamCraft: A Benchmark for Multi-Modal Multi-Agent Systems in Minecraft,” 2024.

[7] A. Vaswani, “Attention is all you need,” *Advances in Neural Information Processing Systems*, 2017.

[8] OpenAI, “GPT-4o,” 2024.

[9] J. Gray, K. Srinet, Y. Jernite, H. Yu, Z. Chen, D. Guo, S. Goyal, C. L. Zitnick, and A. Szlam, “CraftAssist: A Framework for Dialogue-enabled Interactive Agents,” 2019.

[10] S. Rao, W. Xu, M. Xu, J. Leandro, K. Lobb, G. DesGarnes, C. Brockett, and B. Dolan, “Collaborative quest completion with llm-driven non-player characters in minecraft,” *arXiv preprint arXiv:2407.03460*, 2024.