

# インタラクティブフィクションにおける大規模言語モデルの性能

Zhuo Binggang<sup>1</sup> 村田真樹<sup>1</sup><sup>1</sup> 鳥取大学 工学研究科

zhuobinggang@gmail.com murata@tottori-u.ac.jp

## 概要

インタラクティブフィクションにおける人工知能エージェントは主に強化学習に基づいているが、近年の大規模言語モデルの急速な発展を考慮し、我々は大型言語モデルを使用してインタラクティブフィクションタスクを解く方法を提案する。我々が選んだテストデータセットは TextWorld Commonsense (TWC) である。TWC ゲームでは、エージェントのタスクは部屋を整理し、アイテムを適切な位置に配置することであり、タスクを上手く解くにはエージェントが「どのアイテムがどの位置に属するか」という常識を持つ必要がある。提案手法の性能をテストした結果、大規模言語モデルは微調整を受けずに、トレーニングセットで学習された強化学習ベースラインと同等ぐらいの性能を持っていることが分かる。具体的には、GPT4.0 one-shot は Easy および Medium レベルの総計 10 個の TWC ゲームですべてのタスクを完璧に遂行し、Hard レベルのゲームでは、提案手法 (スコアは 0.52) は性能の一番良いベースライン (0.57) に負けたが、この差はプロンプトの改良によって補われる可能性がある。

## 1 はじめに

インタラクティブフィクション (IF)、あるいはテキストアドベンチャーゲームでは、プレイヤーの命令とエンジンの応答がすべて自然言語で行われる。言語を理解するだけでなく、高い得点を獲得するためには、人工知能 (AI) エージェントは他の複雑なスキル、例えば、長期記憶、計画、常識などを持つ必要がある。報酬がスパースであるため、過去の IF の分野における AI エージェントは主に強化学習技術を基にしていたが、近年の大規模言語モデル (LLMs) の急速な発展を考慮し、我々は LLMs を使

用して IF タスクを解く方法を提案する。

我々は提案手法に基づいて GPT 3.5 turbo と GPT 4.0 turbo の性能を TWC データセット [1](詳しい説明は 4.1 節を参照してほしい) でテストした。実験の結果、LLMs は微調整なしで、トレーニングセットで学習された強化学習モデルと競り合う性能を持っていることが示された。

## 2 関連研究

### 2.1 データセット

人間プレイヤー向けに設計された IF は既存の AI エージェント (LLMs を含む) にとってまだ非常に難しい課題であることが分かる [2, 3, 4]。難しい IF から研究するよりも、我々は簡単な IF 環境で AI エージェントの性能を段階的に向上させたいと考えている。適切な難易度を持つ IF 環境を容易く作れるようにするために、Côté らは TextWorld[5] を提案した。TextWorld は、ユーザーが IF 環境を手動で作成するか、生成メカニズムを使用して自動的に IF を構築することを可能にするサンドボックス環境である。TWC[1] は TextWorld 上に構築された IF 環境であり、主な目的はエージェントが常識を活用する能力を検証することである。TWC の後には、より複雑な IF 環境がいくつか登場している。例として、Science World[6] や Complex World[7] がある。Science World の問題点は環境が特定の領域 (科学) に特有であるため、この環境で良いパフォーマンスを示すエージェントは他の環境に対応できない可能性がある。一方、Complex World は TWC よりも複雑であり、Complex World での研究は我々の今後の課題となる。

## 2.2 人工知能エージェント

IF の報酬がスパースであるため、この分野における AI エージェントは主に強化学習に基づいている。LSTM-DQN[8] はこの領域の最初の成果の一つであり、Karthik らは LSTM を使用してテキストの説明を隠れた状態にエンコードした。DRRN[9] は LSTM-DQN をベースにしており、状態とアクションをエンコードするニューラルネットワークを分離する。この改善により、モデルのパフォーマンスと安定性が向上した。KG-A2C[10] は探索の過程で知識グラフを生成し、モデルの探索をさらにガイドする。この方法はエージェントの学習効率とパフォーマンスを向上させたことが報告されている。これらの手法は TWC の論文 [1] でベースラインとして使用されている (LSTM-DQN が LSTM-A2C として実装されていることが要注意)。また、この論文で提案されたモデルである TWC Agent は ConceptNet から常識知識を抽出し、常識知識の導入によりベースラインより良い性能を達成した。

## 3 提案手法

### 3.1 問題設定

TWC ゲームでの人工知能エージェントのタスクは、部屋を整理することであり、具体的には特定の物体をその属する場所に戻すことである。たとえば、使ったティッシュペーパーはゴミ箱に入れるべきであり、汚れた服は洗濯機に入れるべきである。物体が特定の位置に属するかどうかの判断基準は常識であり、これらの物体-位置のペアの妥当性は人間の評価者によって確認されている [1]。

TWC ゲームをプレイする時、各時間ステップで、プレイヤーやエージェントはゲームエンジンからの発見 (observation) を取得する。この発見は自然言語の形で提示される。その後、エージェントは行動を選択し、その行動が実行された後、ゲームエンジンは新しい発見を返す。このサイクルには 2 つの終了条件がある。一つはエージェントがすべてのタスクを完了した場合であり、もう一つはエージェントが行動回数の上限を超えた場合である。本研究では行動回数の上限が 20 に設定される。ゲームエンジンが返す発見には以下の内容が含まれる。

- Action Result: 行動が実行された場合、ゲームエンジンからのフィードバックを意味す

る。行動で報酬を得た場合、その報酬もフィードバックに含まれる。例えば、とあるゲーム状態で「insert dirty yellow T-shirt into washing machine」という行動を実行すると、ゲームエンジンは「You put the dirty yellow T-shirt into the washing machine. Your score has just gone up by one point.」というフィードバックを返す。

- Environment Description: 現在の環境に関するテキストの説明である。
- Inventory: 現在所持しているアイテムの説明である。
- Available Action List: 現在実行できるアクションのリストである。

### 3.2 プロンプト

プロンプトは我々が提案する手法で最も重要な部分である。LLM に提供するプロンプトには、エンジンから受け取った発見の中にあるすべての要素が含まれる: Action Result、Current Environment、Inventory、Available Action List。これらの要素について 3.1 節で説明した。これに加えて、プロンプトにはタスクの説明 (Task)、行動履歴 (Action history)、モデルの出力を構造化するための回答テンプレート (Question、Consideration、Next action) も含まれている。例を以下に示す (長過ぎる文章を省略している)。

Task: You are a experienced text game player, your goal is put things in there proper locations and improve your score.

Action history: Action 0: examine suspended shelf ->The suspended shelf is reliable. Action 1: take dirty singlet ->You pick up the dirty singlet from the ground.

Inventory: You are carrying: a dirty singlet

Current enviroment: -= Laundry Room =- You've entered a laundry room. You begin to take stock of what's here.You can see a washing machine. Empty! .....

Action you can take:

- \* close clothes drier
- \* close laundry basket

.....

- \* put dirty singlet on work table

Question: To put things in there proper locations

and improve your score, what should you do?  
Think step by step then choose 'one' action from  
above list.

Consideration: <fill in>

Next action: <fill in>

「Task」とはタスクの説明で、固定されている文字列である。「Action history」とは行動履歴で、行動の回数とともに増加し、「行動 ->フィードバック」の形式でテキスト化される。「Current Environment」とは現在の環境のテキストの説明である。「Action you can take」とはゲーム環境から取得した可能な行動のリストである。「Question」、「Consideration」および「Next action」は「Task」と同様に固定された文字列であり、モデルの出力を構造化するためのテンプレートである。上記のプロンプトに対する gpt4.0 turbo の出力は以下の通りである。

Consideration: To properly place the dirty singlet, it should be put somewhere related to laundry processing. Inserting it into the washing machine seems like the most appropriate step since typically, dirty clothes need to be washed before they are dried or placed in a laundry basket.

Next action: insert dirty singlet into washing machine

回答テンプレートが存在するため、GPT 4.0 の出力は整然としていて、この出力から行動を自動的に抽出することができる。しかし、GPT 3.5 の出力はテンプレートに合わないことが多いため、出力を行動リストに対応させるためのテクニック (grounding) が必要である。本研究では、LLMs の出力を人手で行動に変換する。具体的には、モデルの出力の一部 (上記の例では「Next action」の後の部分) をコピーし、ゲームエンジンのインターフェースに貼り付けるだけである。

先行研究では、例を提供することが LLMs の性能向上に役立つとされている。そのため、我々はゲームの流れを例としてプロンプトに追加し (「Example walkthrough」と名付ける) 例が一つの場合 (one-shot) を研究した。このゲームの流れは、人間プレイヤーがテストセット外の難易度が hard である一つのゲームでプレイし、最短経路でゲームを完了したものである。「Example walkthrough」の形式は「行動 ->フィードバック」で、「Action history」の形式と完全に一致している。「Example walkthrough」以外のプロンプトは zero-shot と同様

である。one-shot のプロンプトは以下である。

Task: You are a experienced .....

Example walkthrough: Action 0: insert dirty yellow dress into washing machine ->You put the dirty yellow dress into the washing machine.Your score has just gone up by one point. .... Action 14: insert dirty maroon dress into washing machine ->You put the dirty maroon dress into the washing machine.Your score has just gone up by one point. \*\*\* The End \*\*\*

Action history: .....

.....

Next action: <fill in>

### 3.3 API でプロンプトを使用

本節では、API でのプロンプトの使用方法について説明する。我々は OpenAI API を使用して gpt4.0 turbo (具体的なモデル名は gpt-4-1106-preview) および gpt3.5 turbo (モデル名は gpt-3.5-turbo) を呼び出す。Python コードは次の通りである。

```
1 from openai import OpenAI
2 client = OpenAI()
3 completion = client.chat.completions.create(
4     model='gpt-4-1106-preview',
5     messages=[
6         {"role": "system", "content": system_msg
7         },
8         {"role": "user", "content": user_msg}
9     ]
10 )
```

Listing 1 API を使用する Python コード

ここで、*system\_msg* と *user\_msg* は文字列変数であり、本研究ではそれぞれ以下のテキスト内容を含んでいる。

- *system\_msg*: タスク (Task)、ゲームの流れの例 (Example walkthrough)、行動履歴 (Action history)、インベントリ (Inventory)、現在の環境 (Current environment)
- *user\_msg*: 可能な行動リスト (Available Action List)、回答テンプレート (Question、Consideration、Next action)

## 4 実験

### 4.1 データセット

手法の性能を評価するために我々が使用するデータセットは TWC であり、このデータセットは

GitHub から入手できる<sup>1)</sup>。このデータセットには、easy、medium、hard の3つの難易度があり、難易度ごとに5つのゲームが含まれている。ゲームの難易度は整理する必要のあるアイテムや部屋の数で区別する。難易度が最も低いゲームには整理する必要のあるアイテムが1つと部屋が1つしか含まれておらず、難易度が最も高いゲームには整理する必要のあるアイテムが7つと異なる2つの部屋が通路でつながっている。エージェントは「go east」のような行動で部屋間の移動ができる。

## 4.2 実験設定

モデル：GPT4.0 turbo（具体的なモデル名は gpt-4-1106-preview）と gpt3.5 turbo（モデル名は gpt-3.5-turbo）を呼び出すために OpenAI API を使用し、具体的なコードは3.3節で説明した。微調整：すべての LLMs は微調整されていない。行動回数上限：各ゲームについて、モデルが取ることのできる行動の上限を20と規定する。重複実験：重複実験はしていない。すべてのスコア（normalized score）は1回の実行結果である。評価指標：評価指標はTWC論文と同様に normalized score（エージェントが達成されたスコア ÷ 最大の可能なスコア）を使用する。

したがって、モデルと例の数に基づいて、提案手法は次のとおりである：GPT 3.5 zero-shot、GPT 3.5 one-shot、GPT 4.0 zero-shot、GPT 4.0 one-shot。

## 4.3 ベースライン

本研究で使用されるベースラインには、LSTM-A2C、DRRN、KG-A2C、およびTWC Agent（常識知識はNumberbatchで導入）が含まれており、これらのベースラインの特徴について2.2節で紹介した。ベースラインのスコアはTWCの論文から直接取得し、再実装およびテストを行っていないことに注意してほしい。ベースラインのスコアは、10回の繰り返し実験の平均値である [1]。

## 5 結果

実験結果は表1に示す。すべてのスコアは normalized score で、エージェントが達成したスコアをゲームで達成可能な最高スコアで割ったものである。注意すべきなのは、各難易度レベルには5つ

表1 実験結果

	Easy	Medium	Hard
LSTM-A2C	0.86	0.74	0.54
DRRN	0.81	0.73	0.44
KG-A2C	0.85	0.72	0.46
TWC-Agent	0.96	0.85	<b>0.57</b>
GPT3.5 zero-shot	0.86	0.17	0.18
GPT3.5 one-shot	<b>1.00</b>	0.67	0.24
GPT4.0 zero-shot	<b>1.00</b>	<b>1.00</b>	0.49
GPT4.0 one-shot	<b>1.00</b>	<b>1.00</b>	0.52

のゲームが含まれ、スコアはエージェントが5つのゲームで達成したスコアの平均値である。

表1によると、EasyとMediumレベルのゲームでは、GPT4.0（zero-shot および one-shot）が安定してすべてのアイテム整理タスクを遂行した。一方、hardレベルのゲームでは、GPT4.0 one-shot がTWC-Agent および LSTM-A2C に敗れた。ただし、すべてのベースライン手法はトレーニングセットで100エピソードのトレーニングを行っており、LLMsには微調整を行っていない。また、我々が採用したプロンプトは非常に単純なものであり、プロンプトの改良がモデルの性能向上につながる可能性がある。

## 6 おわりに

本研究では、大規模言語モデルを使用してインタラクティブフィクション分野の人工知能エージェントを実現する方法を提案した。従来の強化学習手法と比較して、我々の手法はシンプルで性能が優れている。TextWorld Commonsense データセットでは、微調整を行わなくても、我々の手法はトレーニングセットで学習された強化学習モデルと同等レベルの性能を持っている。具体的には、GPT4.0 one-shot は Easy および Medium の合計10のゲーム環境ですべてのタスクを完璧に遂行した。Hardレベルのゲーム環境では、我々の手法の性能（0.52）は性能の一番良いベースライン（0.57）を上回ることはできないが、この差はプロンプトの改良によって補われる可能性がある。

1) <https://github.com/IBM/commonsense-rl/tree/master/games/twc>

## 参考文献

- [1] Keerthiram Murugesan, Mattia Atzeni, Pavan Kanipathi, Pushkar Shukla, Sadhana Kumaravel, Gerald Tesauero, Kartik Talamadupula, Mrinmaya Sachan, and Murray Campbell. Text-based rl agents with commonsense knowledge: New challenges, environments and baselines. In **Proceedings of the AAAI Conference on Artificial Intelligence**, Vol. 35, pp. 9018–9027, 2021.
- [2] Matan Haroush, Tom Zahavy, Daniel J. Mankowitz, and Shie Mannor. Learning how not to act in text-based games. In **6th International Conference on Learning Representations, ICLR 2018**, 2018. Publisher Copyright: © 6th International Conference on Learning Representations, ICLR 2018 - Workshop Track Proceedings. All rights reserved.; 6th International Conference on Learning Representations, ICLR 2018 ; Conference date: 30-04-2018 Through 03-05-2018.
- [3] Shunyu Yao, Karthik Narasimhan, and Matthew Hausknecht. Reading and acting while blindfolded: The need for semantics in text game agents. **arXiv preprint arXiv:2103.13552**, 2021.
- [4] Chen Feng Tsai, Xiaochen Zhou, Sierra S Liu, Jing Li, Mo Yu, and Hongyuan Mei. Can large language models play text games well? current state-of-the-art and open questions. **arXiv preprint arXiv:2304.02868**, 2023.
- [5] Marc-Alexandre Côté, Akos Kádár, Xingdi Yuan, Ben Kybartas, Tavian Barnes, Emery Fine, James Moore, Matthew Hausknecht, Layla El Asri, Mahmoud Adada, et al. Textworld: A learning environment for text-based games. In **Computer Games: 7th Workshop, CGW 2018, Held in Conjunction with the 27th International Conference on Artificial Intelligence, IJCAI 2018, Stockholm, Sweden, July 13, 2018, Revised Selected Papers 7**, pp. 41–75. Springer, 2019.
- [6] Ruoyao Wang, Peter Jansen, Marc-Alexandre Côté, and Prithviraj Ammanabrolu. Scienceworld: Is your agent smarter than a 5th grader? **arXiv preprint arXiv:2203.07540**, 2022.
- [7] Shreyas Basavatia, Shivam Ratnakar, and Keerthiram Murugesan. Complexworld: A large language model-based interactive fiction learning environment for text-based reinforcement learning agents. In **International Joint Conference on Artificial Intelligence**, 2023.
- [8] Karthik Narasimhan, Tejas Kulkarni, and Regina Barzilay. Language understanding for text-based games using deep reinforcement learning. **arXiv preprint arXiv:1506.08941**, 2015.
- [9] Ji He, Jianshu Chen, Xiaodong He, Jianfeng Gao, Lihong Li, Li Deng, and Mari Ostendorf. Deep reinforcement learning with a natural language action space. **arXiv preprint arXiv:1511.04636**, 2015.
- [10] Prithviraj Ammanabrolu and Matthew Hausknecht.

Graph constrained reinforcement learning for natural language action spaces. **arXiv preprint arXiv:2001.08837**, 2020.

## 7 付録

### 7.1 zero-shot プロンプトの具体例

zero-shot プロンプトの具体例を示す (省略なし)。

Task: You are a experienced text game player, your goal is put things in there proper locations and improve your score.

Action history: Action 0: examine suspended shelf ->The suspended shelf is reliable. Action 1: take dirty singlet ->You pick up the dirty singlet from the ground.

Inventory: You are carrying: a dirty singlet

Current enviroment: -= Laundry Room -=  
You've entered a laundry room. You begin to take stock of what's here.You can see a washing machine. Empty! What kind of nightmare TextWorld is this? You make out an opened clothes drier. The clothes drier is empty! What a waste of a day! You can see a laundry basket. Hmm... what else, what else? The laundry basket is empty! What a waste of a day! You can see a suspended shelf. The suspended shelf is ordinary. But oh no! there's nothing on this piece of garbage. You move on, clearly infuriated by TextWorld. You scan the room, seeing a bench. But the thing hasn't got anything on it. Oh! Why couldn't there just be stuff on it? You make out a work table. But oh no! there's nothing on this piece of garbage.

Action you can take:

- \* close clothes drier
- \* close laundry basket
- \* close washing machine
- \* drop dirty singlet
- \* examine bench
- \* examine clothes drier
- \* examine laundry basket
- \* examine suspended shelf
- \* examine washing machine
- \* examine work table
- \* insert dirty singlet into clothes drier
- \* insert dirty singlet into laundry basket
- \* insert dirty singlet into washing machine
- \* look

- \* put dirty singlet on bench
- \* put dirty singlet on suspended shelf
- \* put dirty singlet on work table

Question: To put things in there proper locations and improve your score, what should you do? Think step by step then choose 'one' action from above list.

Consideration: <fill in>

Next action: <fill in>

### 7.2 one-shot プロンプトの具体例

「Example walkthrough」以外のプロンプトは zero-shot と同様である。文章が長過ぎるため、省略された部分がある。

Task: You are a experienced .....

Example walkthrough: Action 0: insert dirty yellow dress into washing machine ->You put the dirty yellow dress into the washing machine.Your score has just gone up by one point. Action 1: take dirty yellow T-shirt from bench ->You take the dirty yellow T-shirt from the bench. Action 2: insert dirty yellow T-shirt into washing machine ->You put the dirty yellow T-shirt into the washing machine.Your score has just gone up by one point. Action 3: take wet azure dress from suspended shelf ->You take the wet azure dress from the suspended shelf. Action 4: insert wet azure dress into clothes drier ->You put the wet azure dress into the clothes drier.Your score has just gone up by one point. Action 5: take white cap from bench ->You take the white cap from the bench. .... Action 14: insert dirty maroon dress into washing machine ->You put the dirty maroon dress into the washing machine.Your score has just gone up by one point.  
\*\*\* The End \*\*\*

Action history: .....

Inventory: .....

Current enviroment: .....

Action you can take:

.....

Question: .....

Consideration: <fill in>

Next action: <fill in>