Lmgame Bench: Evaluating LLM through Computer Games

Lanxiang Hu*, Mingjia Huo*, Yuxuan Zhang, Haoyang Yu,
Eric P. Xing, Ion Stoica, Tajana Rosing, Haojian Jin, Hao Zhang

HAO AI LAB

جامعة محمد بن زايد للذكاء الاصطناعي
MOHAMED BIN ZAYED UNIVERSITY OF ARTIFICIAL INTELLIGENCE

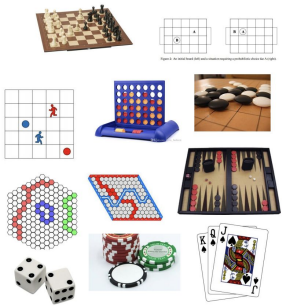
Why do we work on games

Games have been long studied in reinforcement learning.

In the era of LLMs, game is a good testbed for both training and evaluation because:

- Well-defined game **states**, **action** space, and **verifiable rewards**.
- Thousands of gaming env accessible -> easy to **scale**.

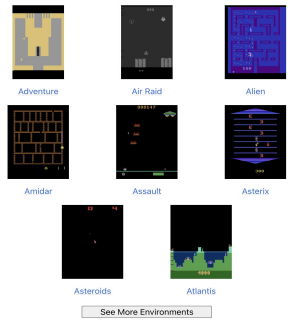
OpenSpiel



Gym

Stable-Retro

Atari Games



Lmgame Bench

Goal 1: Benchmarking

- Game Choice
- Harness design
- Evaluation

Goal 2: Understanding What Games Evaluate

- Training-free analysis
  - Correlation Study
  - Linear model
- Training-oriented analysis
  - Training on Sokoban and Tetris improves diverse tasks.

How do we choose games?

We choose games considering:

1. Evaluate different LLM capabilities => diverse types of games
2. Distinguish model well => moderate or incremental difficulty levels
3. Common games => understandable

Games:

- Board games: Sokoban, 2048, Candy Crush, Tetris
- Real time game: Super Mario Bros
- Detective game: Ace Attorney

Do LLMs play games out-of-box?

No, because:

- **Weak Image understanding**. (demo later on)
- **high latency**, especially for reasoning models.
- Repeated, **uncorrected mistakes**.

-> Need to design agent workflow and harness modes!

Harness Design

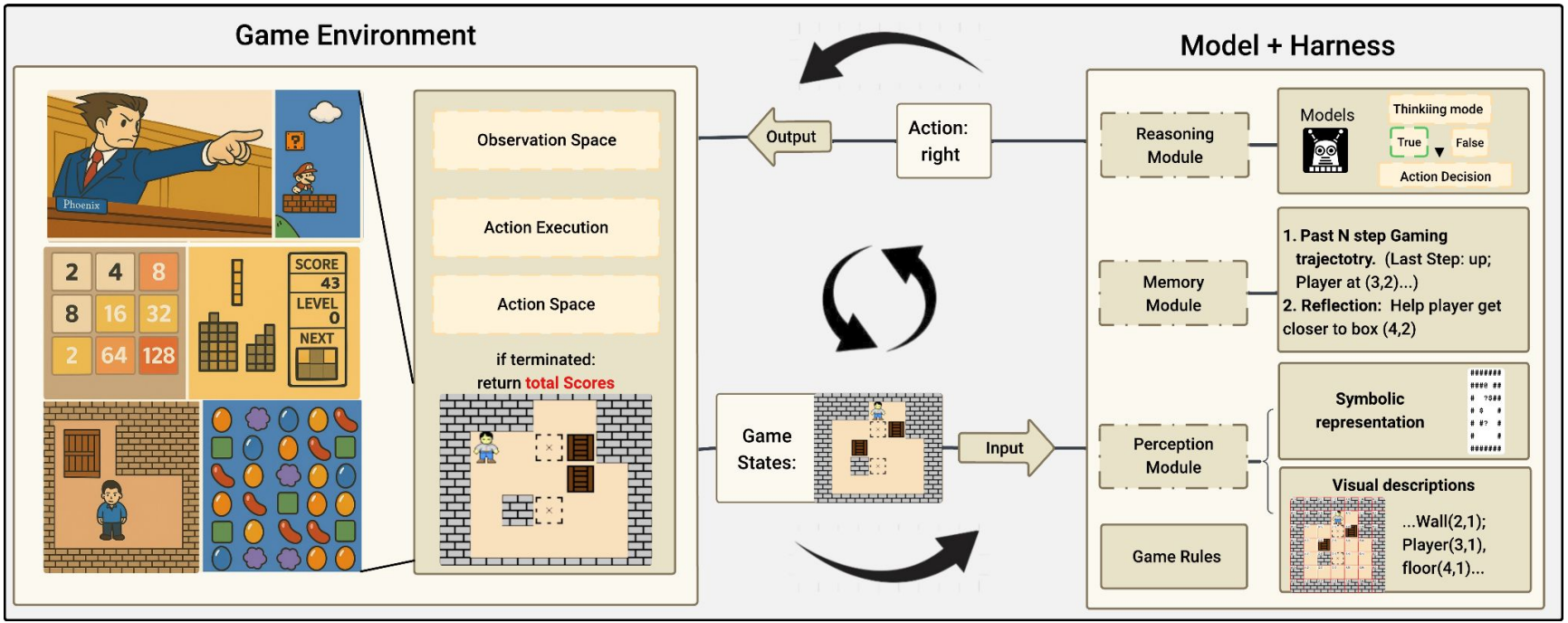
- (Vision) Perception module:
  - Convert image to text:
    - either read game state from backend, or query LLM to convert image into textual description

- Memory module:
  - Short / Long term history: state, action, reward from past turns
  - Reflection: Learns from past mistakes.

- Reasoning module:
  - whether to turn on thinking mode.

Lmgame Bench

- We standardized game environments using Gym and Stable-Retro.



Lmgame Bench

Goal 1: Benchmarking

- Game Choice
- Harness design
- Evaluation

Goal 2: Understanding What Games Evaluate

- Training-free analysis
  - Correlation Study
  - Linear model
- Training-oriented analysis
  - Training on Sokoban and Tetris improves diverse tasks.

Board/Grid games: Sokoban, Tetris, Candy Crush, 2048

- LLMs struggle to understand game boards from only images.
- Gemini shows repeated failures in spatial parsing. Even interpreting a simple board state visually is unreliable.

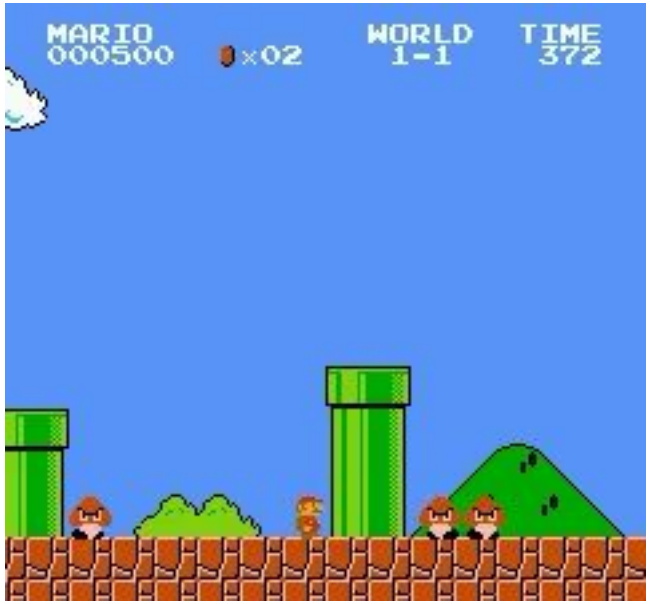
- Our textual formats:
  - 2D ASCII table.
  - object-position list. e.g. Block at (2,3), Player at (4,5)

Real Time Game: Super Mario

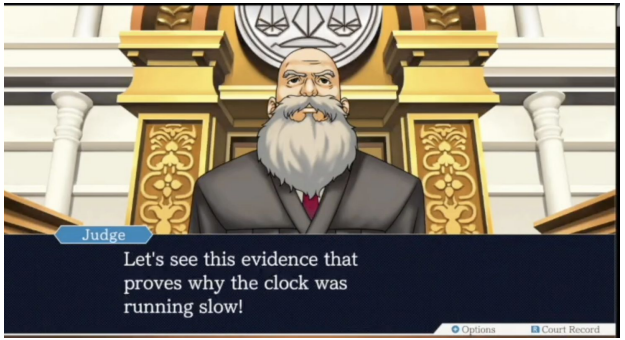
- latency-sensitive:
  - We pause the game until receiving the response
  - Action: (jump, 10), which mean jump for 10 frames.

- Knowing-doing gap:
  - LLM can successfully describe the images, e.g. plan to jump over a pipe
    - LLM thought: "*To safely navigate the pipe, Mario should maintain momentum and height*…."
    - But unable to correctly decide duration of jump, even with game history.

- low FPS problem:
  - LLM only receives one image after the action finished
    - If the character is in the middle air, LLM can't tell



Detective game: Ace Attorney



Ace Attorney is a text-based courtroom game where players examine evidence, question witnesses, and build logical arguments to solve cases.

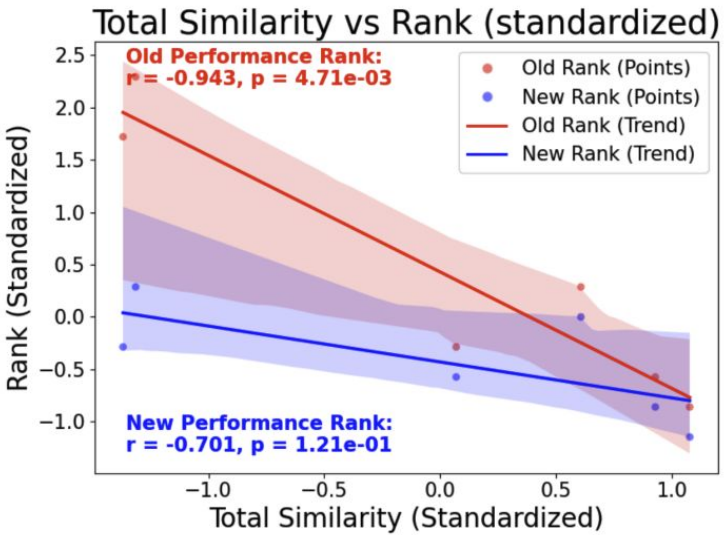
Data contamination detected!

We test whether the models memorized the game or truly reasoning. Using sentence similarity, we find a strong correlation between output similarity and performance.

Text Comparison: Ground Truth vs. o3

Ground Truth Text:	o3 Generated Text:
Frank Sahwit, Round 1 Statement: "I remember the time exactly: It was 1:00 PM." Present: "Cindy's Autopsy Report." Contradiction Exposed: Death occurred "between 4 PM and 5 PM," so the body could not be found at 1 PM.	Frank Sahwit, Round 1 - Statement: "It was exactly 1:00 PM when I saw Larry Butz run out of the apartment." Present: Cindy's Autopsy Report -Contradiction Exposed: Autopsy shows death occurred between 4:00 and 5:00 PM, making a 1:00 PM murder impossible.

Detective game: Ace Attorney



To mitigate data contamination:

- Name replacement
- Prompting to encourage reasoning
- Rewrite background context

Results:

- Before: Strong correlation between transcript similarity and performance → memorization!
- After: Correlation drops → suggests a reduction in memorization effects

Harness Effectiveness

- Gaming harness effectively separates models' performance from random baseline.
- Gaming harness effectively improves models' gaming performance among all games.

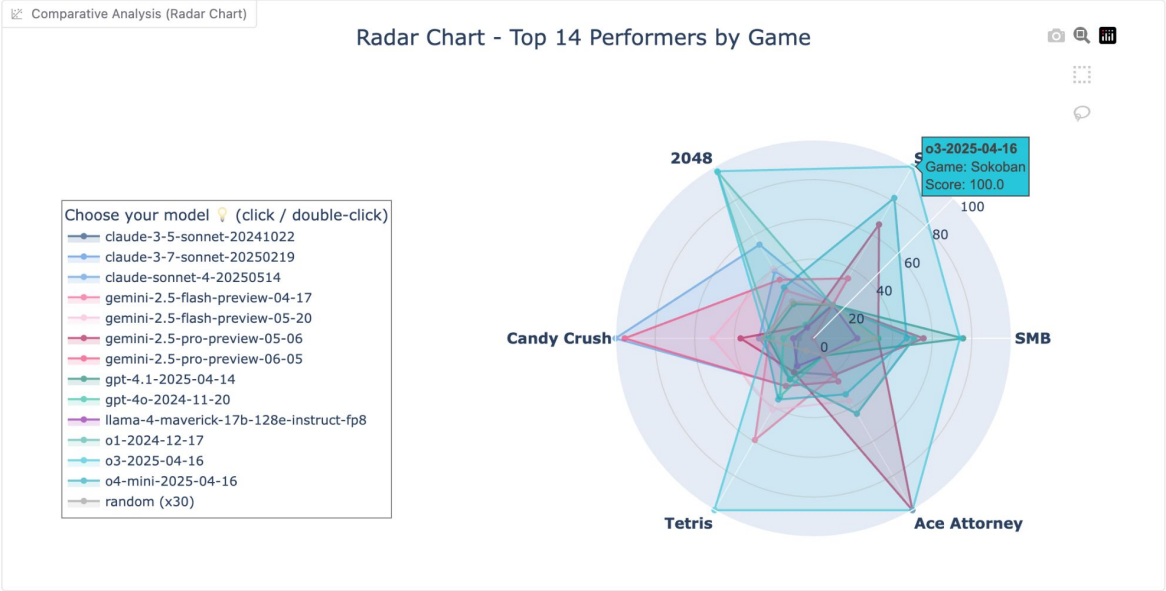
Table 13: Paired-Sample t-Test Results for Harnessed vs. Unharnessed Mean Scores

Game	Δ Mean	$\%\Delta$	$t\,(df = 9)$	p
Candy Crush	+217.50	+224.8%	4.22	0.0022 **
Sokoban	+1.97	+537.5%	3.02	0.0144 *
2048	+17.81	+22.4%	2.36	0.0424 *
Ace Attorney	+3.20	+123.1%	2.36	0.0427 *
Tetris	+5.60	+27.1%	2.27	0.0490 *
Super Mario Bros.	+289.10	+19.3%	1.45	0.1806

* $p < 0.05$, ** $p < 0.01$

Lmgame Bench: Leaderboard

Model Leaderboard: performance without harness.



We also have an **Agent Leaderboard** with all harnesses.

See the details here: https://huggingface.co/spaces/lmgame/lmgame_bench

Lmgame Bench

Goal 1: Benchmarking

- Game Choice
- Harness design
- Evaluation

Goal 2: Understanding What Games Evaluate

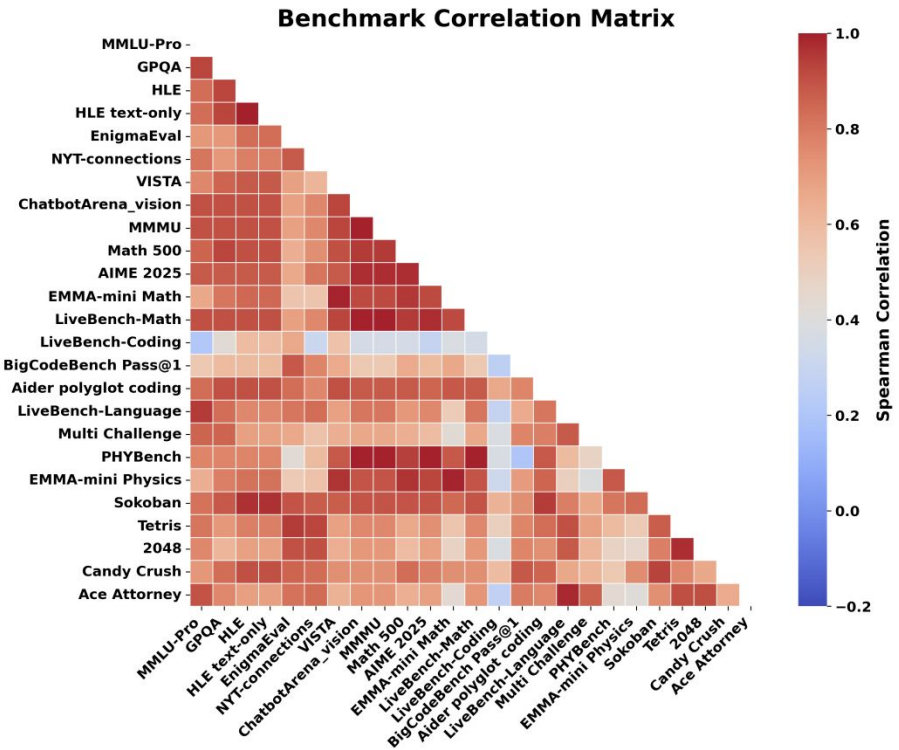
- Training-free analysis
  - Correlation Study
  - Linear model
- Training-oriented analysis
  - Training on Sokoban and Tetris improves diverse tasks.

Gaming benchmark VS other benchmarks

Lmgame-bench shows high **correlation** with other reasoning benchmarks, including math and code.

Method:

- We collect 8 models' rankings on 20 benchmarks
  - *Claude-3.5-Sonnet, Claude-3.7-Sonnet-Thinking, Gemini-2.5-Pro-Preview, Llama-4-Maverick, GPT-4o, o1, o3, and o4-mini*.
- Each benchmark is represented by a vector of model rankings.
- Compute pairwise-correlation among different benchmarks



How individual capabilities contribute to game performance?

Assumption: Game rank is a *linear combination* of ranks on other benchmarks that reflect LLM capabilities.

Physics	EMMA-Physics PHYBench
Math	Math 500 AIME 2025 EMMA-Math LiveBench-Math
Code	BigCodeBench Aider Coding LiveBench-Code
Vision	VISTA MMMU Chatbot Arena (Vision)
Language	MultiChallenge LiveBench-Lang

How individual capabilities contribute to game performance?

Assumption: Game rank is a *linear combination* of ranks on other benchmarks that reflect LLM capabilities.

Results by observing the linear weights:

- Sokoban, Tetris and 2048 ~ Math and Coding.
- Ace Attorney ~ Language
- Super Mario Bros, Candy Crush ~ Physics

Table 3: Learned weights for game ranking prediction using a linear model, where r and RE denote for Pearson's r and mean-normalized residual errors respectively. For the chosen set of categories, the linear models can hardly predict SMB and 2048 rankings correctly.

Game	Language	Physics	Visual	Math	Coding	Offset	r	RE
Sokoban	0.408	1.011	0.810	**2.160**	**2.206**	0.297	0.930	0.4758
Tetris	1.759	0.001	1.356	**1.979**	**2.222**	0.825	0.825	0.814
Ace Attorney	**3.392**	0.000	0.962	2.430	0.004	0.853	0.853	0.800
Super Mario Bros	0.275	**1.905**	0.000	0.597	0.000	**2.940**	0.295	1.377
2048	0.008	0.332	0.000	**1.880**	0.000	**3.130**	0.248	1.467
Candy Crush	0.678	**3.444**	0.002	1.275	2.456	0.088	0.864	0.730

Lmgame Bench

Goal 1: Benchmarking

- Game Choice
- Harness design
- Evaluation

Goal 2: Understanding What Games Evaluate

- Training-free analysis
  - Correlation Study
  - Linear model
- Training-oriented analysis
  - Training on Sokoban and Tetris improves diverse tasks.

Toward Better Agents with Training

- We apply multi-turn RL to train LLMs on Sokoban and Tetris
  - [state 0][response 0][reward 0]...[state i][response i][reward i]...[state n][response n][reward n]
    - Besides a positive reward after progress/success, we also applied a small penalty after each action
  - Response Format: <think>...</think><answer>action || action</answer>
  - Model: Qwen2.5-7B-Instruct
  - 200 steps, batch size = 32
  - Training temperature: 1.0, group_size: 16; Testing temperature: 0
  - PPO with asymmetric clipping (encourage positive reward)
  - Example of Sokoban state:

```
######
######
#O####
#XP###
#__###
######
```

Toward Better Agents with Training

- Trained on Sokoban and Tetris can improve cross-game, planning and spatial reasoning tasks.
  - Blocksworld: Symbolic inputs benefit more than natural language inputs

Table 4: Model performance on diverse tasks after training on simplified Sokoban and Tetris.

Model	Games				Planning			Math/Coding			Agentic
	Sokoban		Tetris		Blocksworld			GSM8K		BIRD	WebShop
	6 × 6	8 × 8	1 type	2 types	Text	1D	2D	1 turn	5 turns	1 turn	
Qwen-7B-Instruct	11.3	5.9	9.0	4.7	64.7	17.9	9.0	**89.5**	**95.3**	**25.0**	7.0
Ours (Sokoban)	**24.2**	**9.0**	17.6	5.1	64.1	**32.7**	**29.5**	89.0	94.1	17.5	**19.1**
Ours (Tetris)	13.3	6.7	**49.5**	**14.5**	**66.2**	21.5	15.2	89.0	93.4	19.8	13.4

Data Mix Training

- Trained on Sokoban -> Sokoban, Blocksworld, Tetris
- Trained on GSM8K -> GSM8K, Tetris
- Trained on mix -> all improved, but not as good as training on a single module

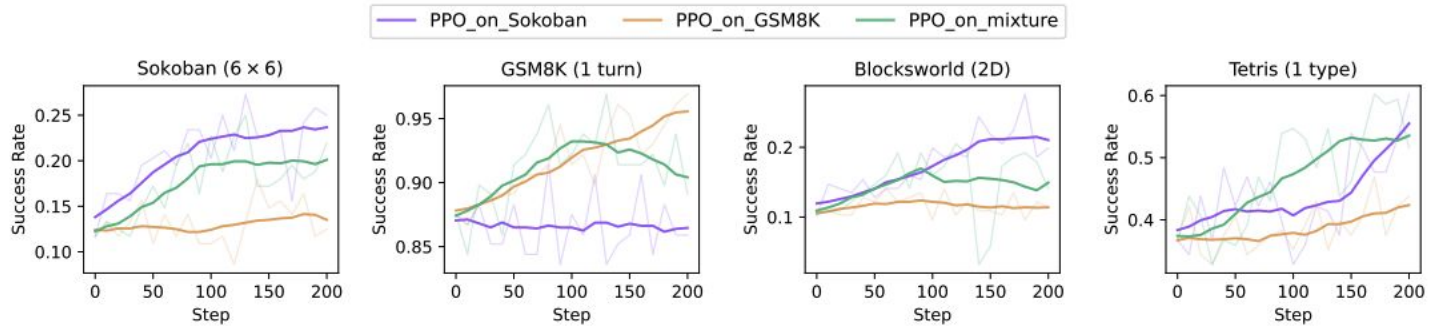


Figure 4: Success rates during training across different evaluation tasks, with models trained on Sokoban, GSM8K, or a half-half mixture of both.

Did we just learn over format, instead of reasoning?

Add **thinking tokens** (<think>...</think>) leads to **significant performance differences** — even before training.

However, models trained on games can outperform both formats (e.g., on Blocksworld-2D), so it's not just learning over the thinking format.

Future work: first SFT on thinking data, then run RL

Table 5: Impact of thinking-token prompting on model generalization across games, planning, and agentic tasks.

Model	Thinking Token		Games				Planning			Agentic
			Sokoban		Tetris		Blocksworld			WebShop
	Train	Test	6 × 6	8 × 8	1 type	2 types	Text	1D	2D	
Qwen-7B-Instruct	-	✓	11.3	5.9	9.0	4.7	64.7	17.9	9.0	7.0
Ours (Sokoban)	✓	✓	**24.2**	**9.0**	17.6	5.1	64.1	**32.7**	**29.5**	**19.1**
Ours (Sokoban)	✗	✓	10.9	8.6	**26.9**	**7.0**	**68.6**	21.2	15.4	10.5
Qwen-7B-Instruct	-	✗	21.1	7.8	1.6	11.3	34.0	17.3	12.8	37.9
Ours (Sokoban)	✗	✗	**24.6**	**12.9**	**34.4**	**19.5**	**44.9**	23.1	13.5	**43.8**
Ours (Sokoban)	✓	✗	19.5	5.1	32.8	12.1	**44.9**	**30.1**	**23.1**	41.8

Bonus: Pokemon Red As An Eval?

Pokemon as an eval has appeared on many latest models' technical reports (e.g. Claude-3.7, Claude-4, Gemini-2.5-pro). Recently we release a blog post on the effectiveness of using the Pokemon game as an eval: https://lmgame.org/#/blog/pokemon_red.

Our takeaway:

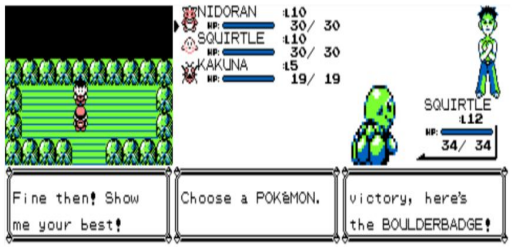


Figure 3: Gemini 2.5-flash wins Boulder Badge without the need for a gaming harness.

Battle control is too easy

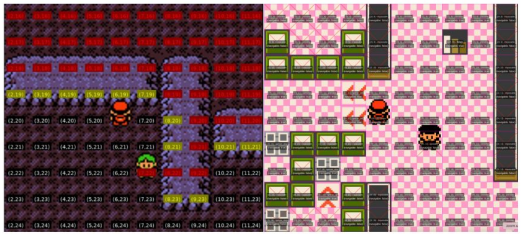


Figure 1: Perception scaffolding implementations for Claude (left) and Gemini (right) with different levels of details.

Navigation with raw game state image is too hard

Model	Steps to reach Oak's Lab	Cost	Time
OpenAI o3	1000	$120	20h
Gemini 2.5 Flash	1000	$50	13h

Table 1: Evaluation Cost for o3 and Gemini-2.5-Flash for the first 1k steps.

Gemini 2.5 report shows the model needing roughly **35,000** actions for a complete gamerun.

Conclusion

- Games offer a rich testbed for evaluating and training AI agents.
- With harnesses, mitigations, and insights into model behavior, Lmgame-Bench reveals the strengths and limitations of LLMs.
- Training-free and training-oriented methods shows that games can both evaluate and potentially improve reasoning.

Potential Future Works:

- SFT
- Tool Calling
- More experiments on LLM / VLM training on games
- LLM vs LLM

Concurrent works that claim generalizability

- SPIRAL: Self-Play on Zero-Sum Games Incentivizes Reasoning via Multi-Agent Multi-Turn Reinforcement Learning
  - Trained on Qwen-7B-base model
  - Format: <think> xxx </think> <answer> xxx </answer>
  - Their reported DeepSeek-Distill-Qwen-7B's baseline number on AIME'24 and MATH500 are both much lower than DeepSeek's official release

- Play to Generalize: Learning to Reason Through Game Play
  - Trained on Qwen-7B-VL-instruct model
  - Evaluated on MathVista and Geometry, and observed generalization.

More About Us

- Our team's missions: we study new perspectives for AI evaluations and the evolving roles humans play in evaluations.
- Current scopes:
  - Enable engaging gameplay while evaluating a variety of large-scale AI models and systems (ICLR'24).
  - Gaming Agents with modular harness design (MAS, ICML'25)
  - Use classical games for AI benchmarking (NeurIPS'25, in submission).
- We are a vibrant and growing team, and we welcome anyone interested in collaborating with us!

X: https://x.com/largemodelgame

Website: https://lmgame.org/

Repo: https://github.com/lmgame-org

Thank you!