THE JOURNAL OF
COGNITIVE SYSTEMS
Journal Homepage: https://dergipark.org.tr/en/pub/jcs

Vol.9, No.2, 2024

Evaluating the Performance of Large Language Models (LLMs) Through Grid-Based Game Competitions: An Extensible Benchmark and Leaderboard on the Path to Artificial General Intelligence (AGI)

[1] Oguzhan Topsakal, [2] Colby J. Edell, [2] Jackson B. Harper

[1] University of South Florida, Tampa, Florida, USA. (e-mail: topsakal@usf.edu)
[2] Florida Polytechnic University, Lakeland, Florida, USA.

ARTICLE INFO

Received: Dec 04th, 2024
Revised: Dec 28th, 2024
Accepted: Dec 29th, 2024

Keywords:
Benchmark
Large Language Model
LLM
LLM Benchmarking
Competition
AI Evaluation
Prompt Engineering
Open Source
Game
Grid Games
Game Simulation
Strategic Thinking
Rule Comprehension
Reasoning
AGI
Artificial General Intelligence

Corresponding author:
Oguzhan Topsakal
✉ topsakal@usf.edu
☎ +1

ISSN: 2548-0650

DOI:https://doi.org/10.52876/jcs.1611181

ABSTRACT

Grid-based games, such as Tic-Tac-Toe, Connect-Four, and Gomoku, offer a valuable platform for evaluating large language models (LLMs) in reasoning, rule comprehension, and strategic thinking which are key skills for advancing Artificial General Intelligence (AGI). Current evaluation benchmarks often focus on tasks like natural language understanding or domain-specific problem-solving, lacking in multi-step reasoning and decision-making assessments. This study introduces an extensible benchmark framework leveraging these games to evaluate LLMs using three prompt types: list, illustration, and image. The framework's modular design facilitates the addition of new games, dynamic rule changes, and advanced prompt engineering techniques, enabling deeper examination of LLM capabilities. Through 2,310 simulated matches, we evaluated leading LLMs, including Claude 3.5 Sonnet, GPT-4 Turbo, and Llama3-70B. Results revealed significant performance variations, with simpler games like Tic-Tac-Toe yielding fewer invalid moves, while more complex games like Connect-Four and Gomoku posed greater challenges. List prompts were generally well-handled, while illustration and image prompts led to higher rates of disqualifications and missed opportunities. The findings underscore the utility of grid-based games as benchmarks for evaluating strategic thinking and adaptability, with implications for robotics, autonomous systems, and interactive AI. Limitations in handling visual data and complex scenarios suggest areas for improvement. The open-source nature of the benchmark encourages transparency and community contributions, fostering collaborative advancements in LLM research. Various future directions are discussed including expanding to novel games to avoid data leakage and contamination issues, increasing the complexity of games, refining prompt techniques, and exploring dynamic rule changes to deepen insights into LLM reasoning capabilities. This study lays the groundwork for advancing AI evaluation through flexible and comprehensive benchmarking tools, guiding progress toward more sophisticated and real-world applications.

1. INTRODUCTION

RECENT advancements in large language models (LLMs) have marked significant progress in the field of Artificial Intelligence (AI) [1]. These developments prompt questions about the potential for achieving Artificial General Intelligence (AGI) [2] and the timeline for such advancements [3]. A critical challenge in the journey towards AGI is developing benchmarks to assess AI's evolving intelligence.
In this study, we introduce a novel and extensible benchmark for LLMs using grid-based games such as Tic-Tac-Toe, Connect Four, and Gomoku, utilizing three distinct types of prompts (list, illustration, image). This benchmark helps assess the capabilities of LLMs, comprising rule comprehension, strategic thinking, and the ability to process and understand complex text and image prompts. Existing benchmarks for LLM evaluation primarily focus on tasks such as natural language understanding, knowledge retrieval, or domain-specific problem-solving. However, these benchmarks fail to adequately measure an LLM's ability to engage in dynamic, multi-step reasoning and decision-making which are critical aspects of general intelligence. By

incorporating grid-based games with varying complexity and requiring adaptation to structured prompts, our benchmark fills this gap and provides a practical, interactive framework for evaluating these underexplored capabilities.

Additionally, it provides a flexible framework for researchers and developers to extend its functionality with minimal programming effort, leveraging built-in game classes and modular code structures. Furthermore, the model could also be used to explore how different prompt engineering techniques impact LLM performance in structured scenarios, comparing effectiveness across text- and image-based formats. It may also be adapted to include dynamic rule changes or novel game mechanics, allowing researchers to test how quickly LLMs can adapt to new constraints. This benchmark offers utility across various roles: algorithm designers and researchers can use it to evaluate LLMs' capabilities and improve strategies for handling structured scenarios; game designers can leverage it to test the adaptability of LLMs to innovative game formats; and players can gain insights into AI performance in strategic and competitive settings, potentially identifying areas for further AI enhancement.

The benchmark provides open-source code for simulating these board games among LLMs and generating data files that store details of the simulated games. We have analyzed and presenting results of a total of 2,310 games played among leading LLMs, consisting of Claude 3 Sonnet and Claude 3.5 Sonnet by Anthropic, Gemini 1.5 Pro and Gemini 1.5 Flash by Google, GPT-4 Turbo and GPT-4o by OpenAI, and Llama3-70B by Meta. The authors encourage contributions by extending the benchmark with new games and welcomes the submission of new results from other LLMs.

2. BACKGROUND AND RELATED RESEARCH

The Transformer architecture was pivotal for the recent rapid developments in natural language processing [4]. This innovation led to the creation of models like BERT (Bidirectional Encoder Representations from Transformers) [5] and OpenAI's GPT (Generative Pre-trained Transformer) series [6]. BERT advanced context understanding by analyzing word relationships within sentences, while the GPT series excelled in generative language capabilities [1]. The scale of LLMs expanded exponentially, resulting in models with billions of parameters and exceptional performance across various NLP tasks. Recent models include GPT-4 by OpenAI [7], Gemini by Google [8], Claude by Anthropic [9], and open-source options like LLaMA by Meta [10]. These models have pushed the boundaries of what is possible with LLMs, showcasing significant advancements in the field. LLMs are employed in diverse tasks such as text summarization, language translation, content generation, and question-answering [11].

2.1. Large Language Model Benchmark

LLMs produce outputs such that their responses can vary even with identical input [12]. Traditional metrics are not suitable for evaluating LLM performance. Instead, specialized datasets and benchmarks are needed to assess LLM capabilities comprehensively [13]. Benchmarks such as GLUE [14], SuperGLUE [15], HELM [16], MMLU [17], BIG-bench [18], ARC [19], TruthfulQA [20], HellaSwag [21], and LiveBench [22] provide diverse tasks that test various aspects of LLMs. GLUE includes tasks like sentiment analysis and question answering to evaluate natural language understanding. Super-GLUE extends GLUE with more demanding tasks such as multi-sentence reasoning and complex reading comprehension. The Massive Multitask Language Understanding (MMLU) benchmark tests LLMs across a wide array of subjects, including mathematics, history, computer science, and law, requiring extensive world knowledge and problem-solving abilities [17]. BIG-bench offers 204 varied tasks in areas such as linguistics, mathematics, reasoning, biology, and software development, allowing researchers to evaluate LLMs comprehensively while managing operational costs [18]. HELM emphasizes transparency and performance in specific tasks, using a multi-metric approach that includes fairness, bias, and toxicity assessments. It continually adapts to add new scenarios, metrics, and models [16]. The AI2 Reasoning Challenge (ARC) from the Allen Institute for Artificial Intelligence assesses AI systems' complex reasoning capabilities through multiple-choice questions. The ARC includes an Easy Set for basic retrieval methods and a Challenge Set for advanced reasoning, pushing AI towards deeper knowledge-based understanding [19]. The TruthfulQA benchmark assesses the accuracy and truthfulness of LLM responses, specifically designed to measure how well models can generate accurate answers and avoid hallucinations [20]. The HellaSwag benchmark tests common sense reasoning by presenting models with sentences and multiple possible endings, requiring them to choose the most logical continuation [21]. LiveBench addresses the test set contamination issue in LLM evaluation by offering a benchmark immune to such contamination and biases from human or LLM judging [22]. It features frequently updated questions from recent sources, automatic scoring against objective ground-truth values, and diverse tasks spanning math, coding, reasoning, language, instruction following, and data analysis [22]. One recent survey reviews three key dimensions: what, where, and how to evaluate [23]. It covers tasks like NLP, reasoning, medical applications, ethics, and education, highlighting the role of evaluation methods in assessing LLM performance and future challenges [23]. Another survey categorizes LLM evaluation into knowledge and capability, alignment, and safety [24], emphasizing rigorous assessment to ensure safe and beneficial LLM development. Both surveys stress the necessity of evaluation to develop proficient and ethically sound LLMs [25].

While these benchmarks provide valuable insights into specific aspects of LLM performance, they often lack comprehensive evaluations of strategic reasoning, long-term planning, and decision-making in dynamic contexts. Additionally, many existing benchmarks do not fully account for the complexities involved in reasoning through real-world problems that require adaptive learning, as seen in tasks like strategic games.

2.2. Utilizing Games for Evaluating LLMs

Existing benchmarks do not assess LLMs' performance in conventional games like chess or Go, which are crucial for evaluating strategic thinking and decision-making. Strategic thinking involves the ability to plan and adapt over multiple steps, while decision-making pertains to selecting optimal actions in dynamic contexts, both of which are critical components of advanced reasoning. Games like chess and Go provide structured environments to test these capabilities, as they require a combination of foresight, adaptability, and complex problem-solving. Incorporating such benchmarks could provide deeper insights into LLMs' potential for tasks demanding sophisticated reasoning and planning. Using games as benchmarks highlights LLMs' abilities to understand rules, formulate strategies, and make decisions. Strategic games test prediction and strategy, while linguistic interaction games test language mastery and context. Text-based games, for example, challenge LLMs to understand natural language, interpret evolving game states, and generate commands within narrative-driven environments, requiring a deep grasp of language and strategy [7]. Studies on text-based games enhance our understanding of LLMs' strengths and weaknesses, laying the foundation for future research to improve their cognitive skills. Such investigations demonstrate the value of using games as benchmarks to expose AI capabilities and limitations, paving the way for developing models with advanced reasoning and strategic thinking.

Studies on models like FLAN-T5, Turing, and OPT in the text-based game "Detective" reveal that these LLMs fall short of state-of-the-art or human performance, struggling with game dynamics, learning from interactions, and goal-oriented processing [26]. A recent study introduces a framework for assessing LLMs through goal-driven conversational games, highlighting their abilities in complex discussions, decision-making, and problem-solving [27]. The SmartPlay benchmark evaluates LLMs across diverse games, focusing on their evolution as intelligent agents [28]. Studies on social interaction games, such as the iterated Prisoner's Dilemma, reveal challenges in strategies requiring mutual understanding and flexibility [29]. Tsai et al. note limitations in LLMs like ChatGPT and GPT-4 in constructing world models and leveraging knowledge in text-based games, suggesting targeted benchmarks [30]. Another study on game theory identifies gaps in LLMs' ability to mimic human rationality [31].

GTBENCH evaluates LLMs' strategic reasoning in 10 game-theoretic tasks covering complete vs. incomplete information, dynamic vs. static, and probabilistic vs. deterministic scenarios [32]. LLMs struggle with complete, deterministic games but perform better in probabilistic ones. Commercial LLMs like GPT-4 outperform open-source models such as CodeLlama-34b-Instruct. Code-pretraining aids strategic reasoning, but advanced methods like Chain-of-Thought (CoT) and Tree-of-Thought are inconsistent. Detailed error profiles are provided [32]. GAMEBENCH evaluates strategic reasoning in LLMs across nine game environments, each highlighting key reasoning skills [33]. Using GPT-3 and GPT-4, along with Chain-of-Thought prompting and Reasoning via Planning (RAP), the study finds improvements, but no model matches human capabilities, with GPT-4 sometimes performing worse than random actions. The games are chosen to avoid overlap with the models' pretraining corpuses.

A recent survey explores the application of LLMs in gaming, identifying their roles, underexplored areas, and future research directions, aiming to reconcile their potential and limitations in the domain [34]. Another survey focuses on LLM-based game agents and their role in advancing AGI, introducing a conceptual architecture centered on perception, memory, thinking, role-playing, action, and learning. It reviews methodologies and adaptation agility across six game genres and suggests future research directions [35].

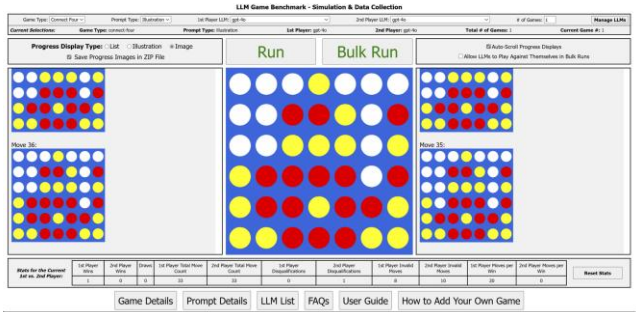
Recent studies have highlighted critical challenges in evaluating LLMs, including Benchmark Data Contamination (BDC) and benchmark dataset leakage. BDC occurs when models are exposed to benchmark-related data during training, skewing evaluation results and impacting generalization. Strategies addressing semantic, information, data, and label-level contamination aim to ensure unbiased assessments [36]. Similarly, benchmark dataset leakage, exacerbated by opaque training practices, has been addressed through a scalable detection pipeline using Perplexity and N-gram accuracy, revealing test set misuse in 31 LLMs. The Benchmark Transparency Card was proposed to encourage clear documentation of benchmark usage, promoting fairness and transparency [37]. Evaluating LLMs through dynamic game competitions may further mitigate BDC and dataset leakage by reducing reliance on static benchmarks and enabling novel, adaptive assessment scenarios.

In a previous study, the authors assessed the strategic decision-making abilities of large language models (LLMs) using Tic-Tac-Toe, leveraging its simplicity and clear outcomes [38]. Across 980 games involving models such as GPT-4, Claude 2.1, and Llama2-70B, GPT-4 demonstrated superior performance with minimal errors across various prompt types. Building on this foundation, the current study significantly expands the scope by providing an open-source benchmark that incorporates three games and three distinct prompt types. A more detailed version of this study has also been published as a preprint on arXiv [39].

3. METHODOLOGY

We have developed a benchmark to evaluate LLMs' capabilities in rule comprehension and decision-making through grid- based games. This benchmark includes open-source web-based software for simulating games, accessible on GitHub [40]. The web application, built with JavaScript, HTML, and CSS, uses server-side AWS Lambda functions in Python to leverage LLMs hosted on AWS Bedrock. The game simulation web app enables LLMs to compete against each other, recording each move's details for analysis in JSON, CSV, TXT, and PNG formats, and summarizing game results. Currently, the benchmark includes Tic-Tac-Toe, Connect Four, and Gomoku, and is designed to be easily extensible to accommodate additional board games with a step-by-step guide for adding new simulations. As shown in Figure 1, the user interface allows users to select a game and the LLMs for the first and second players from a curated list. Users can also choose predefined prompts (e.g., list, illustration, image) and specify the number of consecutive games for the selected combination.

The game simulation initiates by sending the selected prompt to the web API of the chosen LLM for the first player, then awaits its move. Upon receiving a response, the application updates the user interface to reflect the game's progress, as demonstrated in Figure 1, subsequently queries the chosen LLM for the second player, and awaits its move.



LLM Game Benchmark - Simulation & Data Collection

Game Type: Connect Four | Prompt Type: Illustration | 1st Player LLM: gpt-4o | 2nd Player LLM: gpt-4o | # of Games: 1 | Manage LLMs

Current Selections: Game Type: connect-four | Prompt Type: illustration | 1st Player: gpt-4o | 2nd Player: gpt-4o | Total # of Games: 1 | Current Game #: 1

Progress Display Type: ○ List ○ Illustration ● Image

☒ Save Progress Images in ZIP File

Run | Bulk Run

☒ Auto-Scroll Progress Displays

□ Allow LLMs to Play Against Themselves in Bulk Runs

Move 36:

Move 35:

Stats for the Current 1st vs. 2nd Player:	1st Player Wins	2nd Player Wins	Draws	1st Player Total Move Count	2nd Player Total Move Count	1st Player Disqualifications	2nd Player Disqualifications	1st Player Invalid Moves	2nd Player Invalid Moves	1st Player Moves per Win	2nd Player Moves per Win
	1	0	0	33	33	0	1	8	10	20	0

Reset Stats

Game Details | Prompt Details | LLM List | FAQs | User Guide | How to Add Your Own Game

Fig.1. Web app simulates and displays Connect Four game progress.

The prompts, which include the current state of the game, are continuously sent to each LLM's web service until a player wins, the game ends in a draw, or a player is disqualified for making invalid moves. Each query and response are recorded for every move. This methodology ensures seamless interaction between the application and the LLMs via web API calls. The interactions are illustrated in Figure 2.

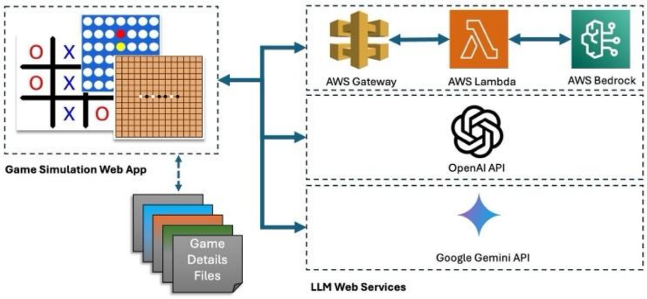


Fig.2. Illustration of web app and web service interactions for game play.

3.1. Games Available and Possibility of Extension

Grid-based games like Tic-Tac-Toe, Connect Four, and Gomoku are ideal for this study due to their structured nature and well-defined rules, which make them effective benchmarks for evaluating strategic reasoning in LLMs. These classical two-player games, played on grids of 3x3, 6x7, and 15x15 respectively, can also be adapted to larger grids. Each is a solved game, meaning the outcome (win, lose, or draw) can be correctly predicted from any position assuming perfect play by both players. In Tic-Tac-Toe, optimal play guarantees a draw, while in Connect Four and Gomoku, the first player can always win with optimal play [41]. These games provide a clear, systematic way to measure an LLM's ability to understand and execute strategies. To encourage further exploration, we have prepared a step-by-step guide for adding new games to the benchmark and welcome contributions.

3.2. LLMs Tested & New Result Submission to the Leaderboard

To ensure a comprehensive assessment, we selected LLMs based on several criteria. We chose models not specifically trained for the benchmark games, assuming proprietary LLMs are not explicitly trained on them. We prioritized high-performing LLMs from industry leaders like OpenAI and Google and included models from notable startups like Anthropic. To enrich our evaluation, we also added the open-source model Meta's Llama3-70B. This selection covers a broad range of innovative approaches, technological capabilities, and accessibility options. Currently, the benchmark includes results and detailed files for the following LLMs: Claude 3.5 Sonnet and Claude 3 Sonnet from Anthropic, Gemini 1.5 Flash and Gemini 1.5 Pro from Google, GPT-4 Turbo and GPT-4o from OpenAI, and Llama3-70B from Meta. These models were selected based on several key criteria. First, they represent state-of-the-art performance, ensuring that the benchmark reflects the most recent advancements in LLM architecture. Additionally, the models span different research organizations and companies (Anthropic, Google, OpenAI, and Meta), providing a broad representation of the landscape of leading LLMs. The selection also includes a variety of architectures and design philosophies, allowing for comparisons across different approaches to LLM development. Finally, the models vary in size, from smaller versions to larger configurations like Llama3-70B, which enables assessment of the impact of model scale on performance. To access these models, we utilized the web APIs provided by Google and OpenAI for the Gemini and GPT-4 models. For accessing models from Meta and Anthropic, we employed Amazon Bedrock services, leveraging serverless AWS Lambda functions and API Gateways, as depicted in Figure 2.

To evaluate the decision-making capabilities of LLMs compared to random play, we included an option to select "random play" as the opponent. This option generates random responses for each move. By testing all the LLMs against random play, we aim to determine the extent to which LLMs outperform random decision-making in game scenarios.

The landscape of LLMs changes rapidly, with new models frequently emerging with improved capabilities. Therefore, we provide game simulation software in the benchmark that generates submission files and encourage new submissions to the leaderboard. Contributors can evaluate other LLMs by integrating their LLM web service URL or API keys, generating new results, and submitting them to the leaderboard. We believe the leaderboard will allow people to see the progress of LLMs in different games as the leaderboard continues to be updated.

3.3. Details of the Prompts

We utilized three types of prompts: list, illustration, and image. The three types of prompts—list, illustration, and image—were selected to vary the presentation of the game state while maintaining consistent structure and content. The 'list' type provides a detailed textual format, dictating occupied spaces by their row and column number. The 'illustration' type visualizes the game state with symbols like 'X', 'O', and 'e' for empty cells, providing a more intuitive representation for the LLM to process. The 'image' type, on the other hand, leverages a visual representation of the game grid, allowing the model to directly interpret the game state from an image format, which may be helpful in environments that require more complex or spatial reasoning. This variety in presentation methods caters to different use cases and ensures that the LLM can handle various forms of input while maintaining consistent prompt structure across different games.

Each prompt is divided into eight main components:

1) an explanation of the game, 2) an explanation of the format for the game status, 3) the current game status, 4) a definition of the LLM's role followed by a request for its next move, 5) an explanation of the response format, 6) an explanation of invalid moves, 7) a warning if the previous move was invalid, including an explanation of why it was deemed invalid, and 8) the current number of invalid moves made by the player, as

well as the number of invalid moves until the player is disqualified. The current game status, the invalid move warning, and the invalid move counts are dynamically generated and updated as the game progresses. Table I presents the components of a ’list’ type prompt for the Tic-Tac-Toe game. This standardized format ensures consistency in prompts throughout the game while allowing for dynamic updates of the game state.

TABLE I
THE PARTS OF A LIST PROMPT FOR TIC-TAC-TOE. THE DIFFERENCES IN THE ‘ILLUSTRATION’ AND ‘IMAGE’ PROMPTS ARE GIVEN IN TABLE II.

Part	Prompt Content
The explanation of the game. Similar style explanation is given for other games.	Tic-Tac-Toe is a two-player game played on a 3 by 3 grid. The first player uses X symbols, and the second player uses O symbols. Players take turns placing their symbols in an empty cell on the grid. The objective is to align three of your symbols either horizontally, vertically, or diagonally. The player who first aligns three of their symbols wins the game. Strategic placement is crucial; besides aiming to align their symbols, players must also block their opponent's potential alignments to avoid defeat.
The explanation of the format for the status of the game. The same for every game for the selected prompt type. The sample on the right is for the ‘list’ type of prompt.	The current state of the game is recorded in a specific format: each occupied location is delineated by a semicolon (';'), and for each occupied location, the row number is listed first, followed by the column number, separated by a comma (','). If no locations are occupied by a player, 'None' is noted. Both the row and column numbers start from 1, with the top left corner of the grid indicated by 1,1.
The current game status. The sample on the right shows the current state for the ‘list’ type of prompt.	The current state of the game is as follows: The locations occupied by the first player: 1,1; 1,2; 3,2. The locations occupied by the second player: 2,2; 3,3.
Defining the role of the LLM and then asking its next move. The same for every game.	You are an adept strategic player, aiming to win the game in the fewest moves possible. You are the first (second) player. What would be your next move?
The explanation of the response format.	Suggest your next move in the following JSON format: *{*'row': RowNumber, 'column': ColumnNumber*}*. Do not include any additional commentary in your response. Replace RowNumber and ColumnNumber with the appropriate numbers for your move. Both RowNumber and ColumnNumber start at 1 (top left corner is *{*'row': 1, 'column': 1*}*). The maximum value for RowNumber and ColumnNumber is 3, as the grid is 3 by 3.
The explanation of the invalid moves.	Please note that your move will be considered invalid if your response does not follow the specified format, or if you provide a RowNumber or ColumnNumber that is out of the allowed range, or already occupied by a previous move. Making more than 3 invalid moves will result in disqualification.
The warning if the last move was invalid, including a copy of the previous move and an explanation of why the move was invalid.	Your previous response was '{”row”: X, “column”: Y*}*'. This move was deemed invalid for the following reason: 'Already Taken'. Please adjust accordingly.
The current number of invalid moves, as well as the number of invalid moves left until disqualification.	You currently have X invalid move(s). Y more invalid moves will result in disqualification.

The content of the three types of prompts is consistent, except for the representation of the current state of the game (previous moves). The ‘list’ prompt enumerates previous moves for each player in a “row, column” format. The ‘illustration’ prompt depicts the current state of the grid using specific symbols for the first and second players (X and O for Tic-Tac-Toe, R and Y for Connect Four, and B and W for Gomoku) and ’e’ for empty cells. The ‘image’ prompt visualizes the current state by providing a snapshot of the game board. The structure of the game status and formatting for the ‘list’ prompt type are provided in Table I. The differences in the game status and formatting in the ‘illustration’ and ‘image’ prompt types are detailed in Table II.

LLMs use parameters like max tokens, temperature, top-p, and frequency penalty [42]. Max tokens control length, temperature adjusts creativity, top-p limits word choices to balance creativity and coherence, and frequency penalty reduces repetition. These settings customize LLM responses for applications such as customer support and content creation. We use default configurations for all parameters except the prompt, trusting the creators’ settings for optimal performance.

The games continued until one player won, a draw occurred, or a disqualification was necessary. To gather statistical data, each game was repeated five times. Disqualification occurred if a player made more than the allowed number of invalid moves: three for Tic-Tac-Toe, six for Connect Four, and fifteen for Gomoku. A move was invalid if it didn’t follow the specified format, the RowNumber or ColumnNumber was out of range, or the space was already occupied. Players were warned and given reasons for invalid moves (as shown in Table I) and asked to retry. Continuous invalid moves led to disqualification to prevent delays. To ensure fairness and avoid skewing the results due to an opponent’s disqualification, a player’s disqualification did not count as a win for the other player.

TABLE II
DIFFERENCES IN THE ILLUSTRATION AND IMAGE PROMPTS FOR TIC-TAC-TOE ARE OUTLINED BELOW. CONNECT FOUR AND GOMOKU PROMPTS FOLLOW A SIMILAR STYLE.

Part	Prompt Content
The explanation of the format for the status within the *illustration* prompt.	The current state of the game is illustrated on a 3 by 3 grid. 'X' represents positions taken by the first player and 'O' represents positions taken by the second player, while 'e' indicates an empty (available) position.
The current game Status within the *illustration* prompt.	The current state of the game is as follows: eXe eeO eOe
The explanation of the format for the status within the *image* prompt.	The current state of the game is depicted in an image showing a 3 by 3 grid, where 'X' represents positions taken by the first player and 'O' represents positions taken by the second player.
The current game Status within the *image* prompt.	The current state of the game is given in the attached image. [Image is sent in base64 format]

During the game sessions conducted through the web application, data on gameplay was collected and stored in JSON, CSV, TXT, and PNG formats. Samples and complete data from bulk runs are available on GitHub. The JSON files include details such as date/time, players, game results, duration, and all moves (both valid and invalid), along with the game status sent to the LLM and the responses received. JSON is also used for leaderboard submissions. A CSV file provides a streamlined summary of the game, the TXT file offers an illustrated representation of the moves, and PNG files display snapshots of the board after each move. All generated files are publicly available on GitHub [40].

The charts and tables are prepared using data files generated by the open-source game simulation web software, shared on GitHub. These files, containing data from 2,310 games, are also available on GitHub [40].

4. RESULTS

4.1. Tic-Tac-Toe: Wins, Draws, and Disqualifications

Figure 3 displays the outcomes of 280 Tic-Tac-Toe games using the list prompt type, where seven LLMs and a random play opponent engaged in five matches per opponent. The chart summarizes win, draw, and disqualification rates for each LLM and random play as both the first and second players. Claude 3.5 Sonnet has the highest win rate as the first player (88.57%) but a lower win rate as the second player (17.14%). GPT-4o and Gemini 1.5 Pro perform well as both first and second players, while random play results in the highest disqualification rates.

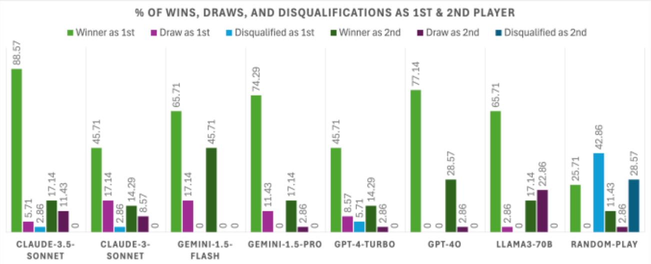


Fig. 3. Tic-Tac-Toe outcomes using the 'list' prompt: each LLM played 5 games as both player 1 and 2 against six others and "random play" (280 games total).

Figure 4 shows the performance metrics of the seven LLMs and random play using the illustration prompt format. Claude 3.5 Sonnet has the highest win rate as the first player, demonstrating a strategic advantage. Llama3-70B and GPT- 4 Turbo also perform well. Some models, like Gemini 1.5 Flash, have notable disqualification rates due to occasional invalid moves. The random player has lower win rates and higher disqualification rates, underscoring the LLMs' superior strategic capabilities.

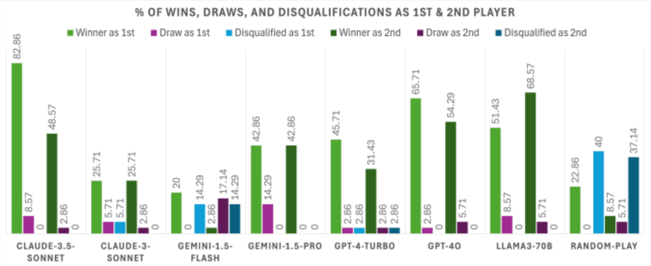


Fig. 4. Tic-Tac-Toe outcomes using the 'illustration' prompt: each LLM played 5 games as both player 1 and 2 against six others and "random play" (280 games total).

Figure 5 presents the performance metrics using the image prompt format. Claude 3.5 Sonnet shows high disqualification rates as both the first (46.67%) and second player (53.33%), indicating rule compliance issues. GPT-4 Turbo and Gemini 1.5 Pro also have significant disqualification rates. GPT-4 Turbo and GPT-4o exhibit the highest win rates. The random play baseline has high disqualification rates, highlighting the strategic advantages of LLMs. No draws occurred with the image prompt, and Llama3-70B was excluded from image prompt tests as it does not accept images.

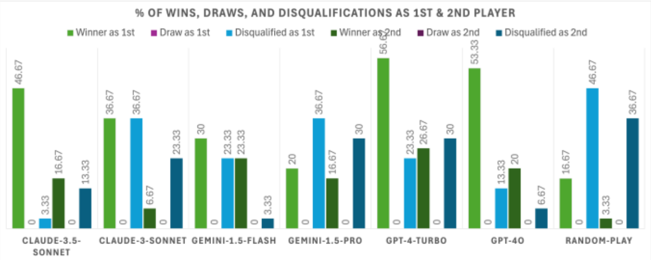


Fig. 5. Tic-Tac-Toe outcomes using the 'image' prompt: each LLM played 5 games as both player 1 and 2 against five others and "random play" (210 games total).

4.2. Connect-4: Wins, Draws, and Disqualifications

Figure 6 displays the performance metrics of various LLMs and a random play strategy in Connect Four using the list prompt type. Claude 3.5 Sonnet and Gemini 1.5 Pro show outstanding performance with an 88.57% win rate as the first player. Most LLMs demonstrated strong overall win rates as both first and second players. The random player, serving as a baseline, has lower win rates and some disqualifications, highlighting the LLMs' strategic advantages. No draws occurred in these games.

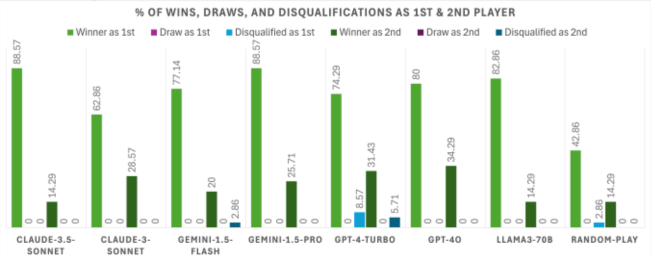


Fig. 6. Connect Four outcomes using the 'list' prompt: each LLM played 5 games as both player 1 and 2 against six others and "random play" (280 games total).

Figure 7 presents the performance metrics using the illustration prompt type. GPT-4 Turbo has the highest disqualification rates as both first and second players. The random play baseline has the second-lowest win rate as the first player and the lowest as the second player. No draws occurred in these games.

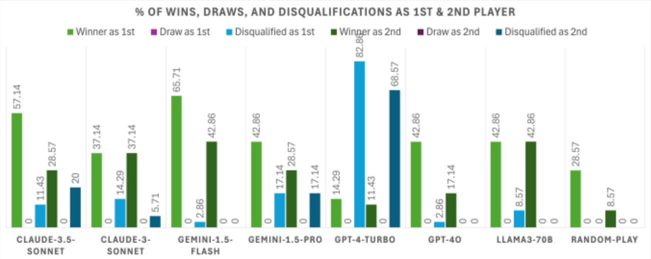


Fig. 7. Connect Four outcomes using the 'illustration' prompt: each LLM played 5 games as both player 1 and 2 against six others and "random play" (280 games total).

Figure 8 illustrates the performance metrics using the image prompt format. GPT-4 Turbo and Claude 3.5 Sonnet demonstrate strong winning performance as both first and second players. Claude 3 Sonnet and Gemini 1.5 Flash have high disqualification rates. The random play baseline has the lowest win rates. No draws occurred during these matches.

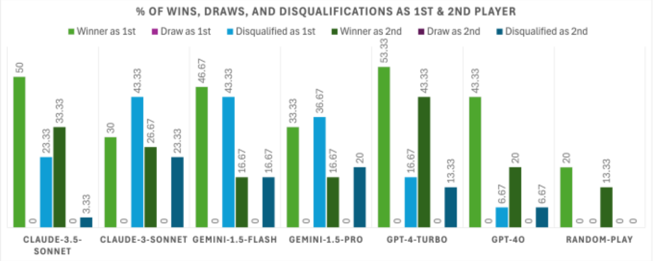


Fig. 8. Connect Four outcomes using the ‘image’ prompt: each LLM played 5 games as both player 1 and 2 against five others and “random play” (210 games total).

4.3. Gomoku: Wins, Draws, and Disqualifications

Figure 9 displays the performance metrics of various LLMs and a random play strategy in Gomoku using the list prompt. Claude 3.5 Sonnet demonstrates exceptional performance with a 94.29% win rate as the first player and 25.71% as the second player, with no disqualifications. Other models also perform well, but Gemini 1.5 Pro, GPT-4 Turbo, and GPT-4o have some disqualifications. The random player baseline has no wins, draws, or disqualifications.

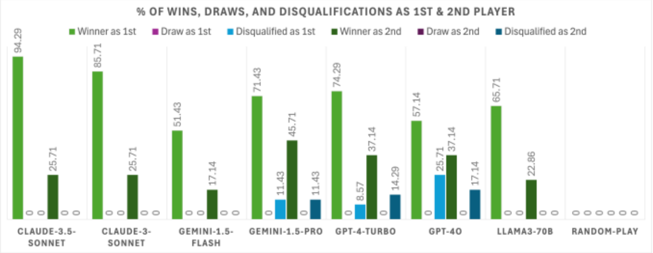


Fig. 9. Gomoku outcomes using the ‘list’ prompt: each LLM played 5 games as both player 1 and 2 against six others and “random play” (280 games total).

Figure 10 shows the performance metrics in Gomoku using the illustration prompt. Significant disqualification rates are observed, particularly for second players, indicating rule adherence challenges. Models like Gemini 1.5 Flash and Llama3- 70B had high disqualification rates. Win rates vary widely, with some models performing well as the first player but struggling as the second. The notable disqualification rates suggest that strategic complexity and rule comprehension are significant factors. The random player baseline, with no wins, draws, or disqualifications, underscores the superior strategic thinking of the LLMs despite their challenges.

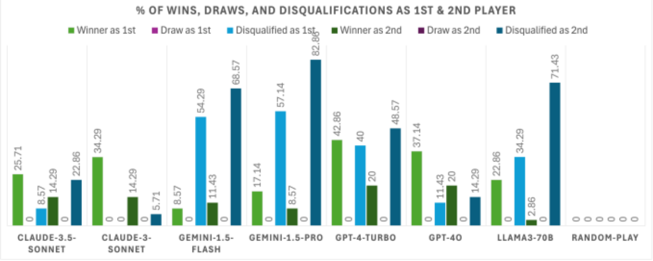


Fig. 10. Gomoku outcomes using the ‘illustration’ prompt: each LLM played 5 games as both player 1 and 2 against six others and “random play” (280 games total).

Figure 11 illustrates performance metrics using the image prompt type. High disqualification rates are noted for some models, especially as the second player, indicating rule adherence difficulties. Win rates vary significantly, with some models achieving higher success as the first player. The random player baseline shows no wins, draws, or disqualifications, highlighting the superior performance of the LLMs.

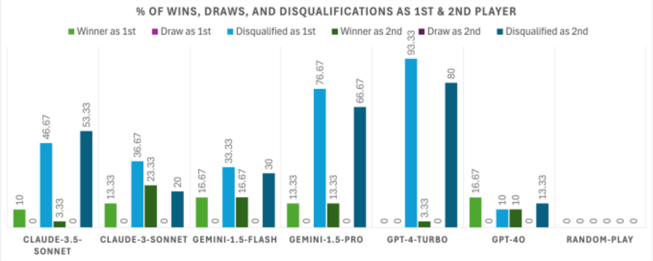


Fig. 11. Gomoku outcomes using the ‘image’ prompt: each LLM played 5 games as both player 1 and 2 against five others and “random play” (280 games total).

4.4. Tic-Tac-Toe: Move Analysis

The chart in Figure 12 illustrates the performance of LLMs and a random play strategy in terms of moves per game and invalid moves per game in Tic-Tac-Toe across three prompt types: list, illustration, and image. The random play serves as a baseline, indicating performance without strategic thinking. The number of moves per game increases with the prompt’s complexity, with image prompts resulting in the highest moves. For list prompts, LLMs ranged from 6.46 to 7.43 moves per game, while random play had 10.11. Illustration prompts saw a slight increase, peaking with Gemini 1.5 Flash. Invalid moves were minimal for list prompts but increased for illustration and were highest for image prompts, particularly for Claude 3 Sonnet and Gemini 1.5 Pro. Random play exhibited higher invalid moves across all prompt types, underscoring its lack of strategic planning. Llama3-70B was not used for the image prompt as it cannot accept images.

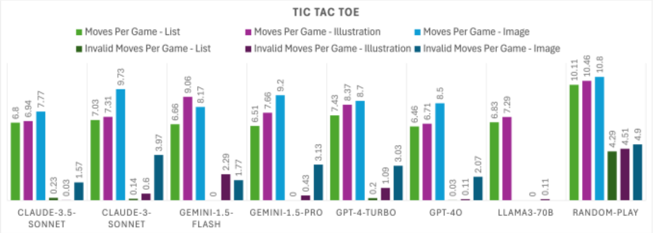


Fig. 12. Moves per game and invalid moves (already taken) per game for Tic-Tac-Toe.

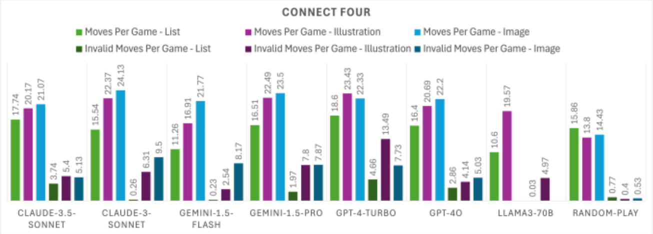


Fig. 13. Moves per game and invalid moves (already taken) per game for Connect Four.

4.5. Connect-Four: Move Analysis

The chart in Figure 13 compares the performance of LLMs and random play in Connect Four across three prompt types. Moves per game consistently trended toward the fewest moves with the ‘list’ prompt type, and the largest amount of moves with the ‘image’ prompt type. For list prompts, LLMs showed consistent moves with minimal invalid moves. Invalid moves increased significantly for GPT-4 Turbo with illustration prompts and for Claude 3 Sonnet with image prompts. Random play showed relatively lower invalid moves, likely due to the nature of Connect Four. These results highlight the challenges LLMs face with more complex prompts.

4.5. Gomoku: Move Analysis

Figure 14 illustrates the performance of LLMs and random play in Gomoku across the three prompt types. Moves per game across prompt formats followed a similar trend to the Tic-Tac-Toe results. Invalid moves were minimal in list prompts but increased for illustration and image prompts, especially for Gemini 1.5 Flash, GPT-4 Turbo, and Llama3-70B. Random play showed fewer invalid moves, as it is less likely to place a move on an already occupied space early in the game on the 15x15 grid. This chart highlights the challenges LLMs face with more complex prompts.

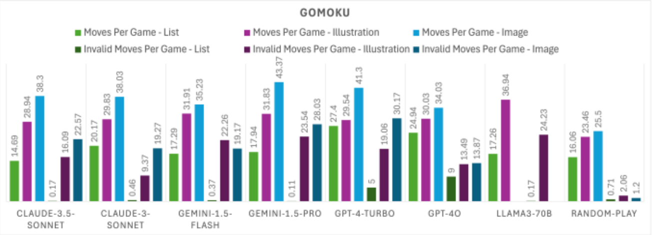


Fig. 14. Moves per game and invalid moves (already taken) per game for Gomoku.

We analyzed LLMs' strategic decision-making by counting missed opportunities to win or block an opponent's win with one move. For example, in Tic-Tac-Toe, if a player had two symbols in a row and did not place its next move to win or block, it was counted as a missed opportunity. Our analysis covered 70 games per LLM for list and illustration prompts and 60 games per LLM for image prompts. Fewer missed opportunities per game indicate better performance. However, LLMs making many invalid moves and getting disqualified without creating opportunities to win could falsely show zero missed opportunities. To avoid confusion, we normalized missed opportunities by valid moves, calculating the percentage of opportunities missed per valid move.

4.6. Tic-Tac-Toe: Missed Wins/Blocks Analysis

Figure 15 shows the percentage of missed win and block opportunities per valid move for various LLMs in Tic-Tac-Toe across list, illustration, and image prompts. Blue bars represent missed win opportunities, and orange bars represent missed block opportunities. Generally, LLMs missed fewer opportunities with list prompts compared to illustration and image prompts. For instance, Claude 3.5 Sonnet had a 9% missed win rate and a 20% missed block rate for list prompts, which increased to 21% and 28% for illustration prompts. GPT-4 Turbo also showed a notable increase in missed block opportunities with illustration prompts. The trend indicates that LLMs struggle more with visually complex prompts, leading to higher missed strategic opportunities. The chart highlights that LLMs can identify winning moves but often struggle more with blocking opponents' winning moves in more visual-based prompt types such as 'illustration' and 'image'.

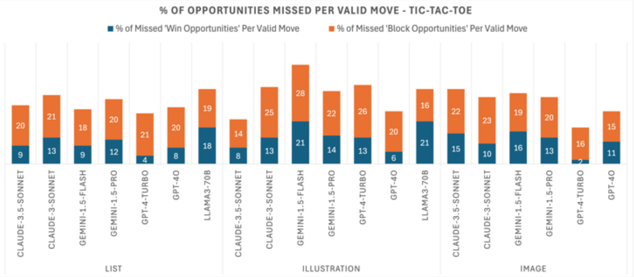


Fig. 15. Percentage of strategic move opportunities missed per valid Tic-Tac-Toe move.

4.7. Connect Four: Missed Wins/Blocks Analysis

Figure 16 shows the percentage of missed strategic move opportunities per valid move in Connect Four. For list prompts, Claude 3 Sonnet missed the most block opportunities. In illustration prompts, Llama3-70B and GPT-4 Turbo performed better. In image prompts, missed block opportunities were notably higher for Claude 3.5 Sonnet and Gemini 1.5 Pro.

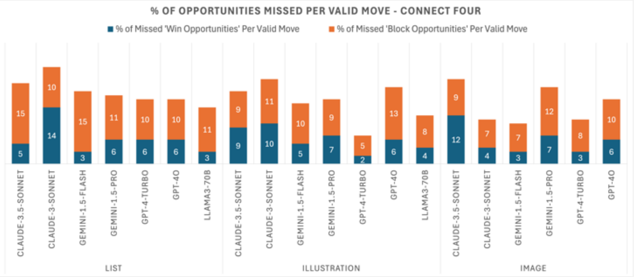


Fig. 16. Percentage of strategic move opportunities missed per valid Connect Four move.

4.8. Gomoku: Missed Wins/Blocks Analysis

Figure 17 displays the percentage of missed win and block opportunities per valid move in Gomoku across the three prompt types. LLMs generally show a higher percentage of missed block opportunities compared to win opportunities. For list prompts, Gemini 1.5 Flash missed the most block opportunities. For illustration prompts, Gemini 1.5 Pro had no missed opportunities. For image prompts, GPT-4 Turbo minimized missed opportunities best.

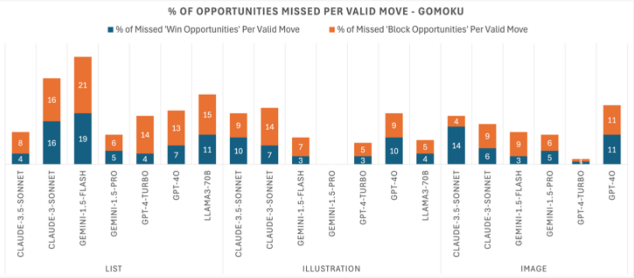


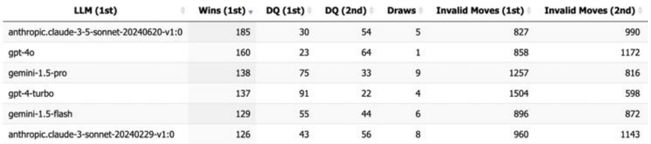
Fig. 17. Percentage of strategic move opportunities missed per valid Gomoku move.

Further analysis can be conducted using the Game Simulation web app or the open-access data on GitHub. For example, analyzing the creation of winning opportunities can provide additional insights. Contributions to the repository with suggestions and new evaluation metrics are encouraged to enhance the assessment of LLM capabilities in grid-based games.

4.9. Leaderboard

Figure 18 displays a portion of the leaderboard page of the LLM Game Benchmark, summarizing the results of games played between various LLMs and a random play generator. Key metrics on the leaderboard include win ratios, number of wins, disqualifications, invalid moves, and total moves for both the first and second players. Users can filter games by type, prompt type, and LLM players, and sort results by clicking on any column header. The leaderboard page also allows aggregation of results by game type, prompt type, and LLM player.

The initial data includes results from Claude 3.5 Sonnet, Claude 3 Sonnet, Gemini 1.5 Flash, Gemini 1.5 Pro, GPT-4 Turbo, GPT-4o, and Random-Play. Each LLM played against every other LLM five times for each game and prompt type combination. New game result submissions are welcome and can be generated using the game simulation web software. The leaderboard web page can be accessed on the benchmark's GitHub page [41]. The data used to populate the leaderboard can be downloaded in JSON format.



LLM (1st)	Wins (1st)	DQ (1st)	DQ (2nd)	Draws	Invalid Moves (1st)	Invalid Moves (2nd)
anthropic.claude-3-5-sonnet-20240620-v1:0	185	30	54	5	827	990
gpt-4o	160	23	64	1	858	1172
gemini-1.5-pro	138	75	33	9	1257	816
gpt-4-turbo	137	91	22	4	1504	598
gemini-1.5-flash	129	55	44	6	896	872
anthropic.claude-3-sonnet-20240229-v1:0	126	43	56	8	960	1143

Fig. 18. A snapshot from the leaderboard showing aggregated results.

5. DISCUSSION

This study provides a comprehensive evaluation of seven LLMs, Claude 3.5 Sonnet, Claude 3 Sonnet, Gemini 1.5 Flash, Gemini 1.5 Pro, GPT-4 Turbo, GPT-4o, and Llama3-70B alongside a random play generator across three games (Tic-Tac-Toe, Connect Four, and Gomoku) and three prompt types (list, illustration, image). Each LLM played five games per game and prompt type against each opponent, resulting in a total of 2,310 games for analysis.

LLM performance varied significantly across different games and prompt types. Simpler games like Tic-Tac-Toe had fewer invalid moves and disqualifications compared to more complex games like Connect Four and Gomoku, highlighting the models' varying capacities to handle increased game complexity. LLMs performed best with list prompts for Tic-Tac-Toe and Connect Four, while illustration and image prompts led to higher disqualification rates and missed strategic opportunities, indicating difficulties in interpreting visual data and maintaining consistency.

The random play strategy consistently recorded the highest number of losses and invalid moves, serving as a useful baseline. The stark contrast between random play and the LLMs underscores the models' capacity for strategic decision-making, though improvements are needed in handling complex and visual data.

In Tic-Tac-Toe and Connect Four, LLMs performed well with list prompts, exhibiting minimal invalid moves. Performance declined with illustration prompts, and the most significant decline was with image prompts, where many LLMs displayed a high number of invalid moves. Connect-Four showed increased invalid moves compared to Tic-Tac-Toe, reflecting higher complexity. For Gomoku, performance was mixed across all prompt types, with a notable increase in invalid moves and disqualifications, indicating significant interpretative and decision-making challenges.

Prompt type significantly impacted LLM performance. List prompts were generally well-handled, suggesting textual representation of game states is within their capabilities. Illustration prompts posed moderate challenges, and image prompts were the most challenging, highlighting a need for improvement in processing image-based inputs.

Invalid move analysis revealed no out-of-bounds errors, likely due to updated prompts that clearly defined possible column and row values, especially compared to our previous study. Invalid format errors, mostly due to hallucinated tag names in JSON, were made only by GPT- 4 Turbo and Gemini 1.5 Flash. A significant percentage of invalid moves were due to moving to an already occupied space.

The analysis of missed strategic opportunities highlighted variability in decision-making processes. Models like Claude 3.5 Sonnet and GPT-4 Turbo showed fewer missed opportunities with list prompts but struggled with illustration and image prompts.

The findings have broader implications for LLM applications in fields such as robotics, autonomous systems, and interactive AI. Improving LLMs' strategic thinking and decision making abilities can enhance performance in real-world tasks requiring similar cognitive skills.

The extensible nature of the benchmark, with its modular code and open invitation for community contributions, represents a significant step towards collaborative LLM research. Encouraging researchers to add new games and share results can lead to a more dynamic and comprehensive evaluation framework. Future work can include a broader range of games and tasks to evaluate LLMs across different strategic environments.

6. LIMITATIONS and FUTURE DIRECTIONS

The study primarily focuses on grid-based games, which, while useful, may not fully capture the breadth of real-world strategic interactions. Future benchmarks should incorporate a wider variety of game types, including those with more complex rules and long-term strategic planning, to provide a more comprehensive assessment of LLM capabilities.

We made the assumption that the LLMs were not specifically trained to play the evaluated grid-based games, relying instead on their general ability to process and understand game rules through the provided prompts. However, it is difficult to completely rule out the possibility that these models may have encountered similar games or strategies during training. This introduces a potential bias, as the LLMs may simply be mimicking known moves rather than genuinely comprehending the game rules. To address this limitation, future studies could evaluate LLMs using existing games with altered rulesets, or entirely new games. By introducing novel rules or variants, researchers can ensure that the LLMs are comprehending the game dynamics rather than relying on memorized patterns from training data.

Designing custom games to test specific aspects of LLM capabilities, such as adapting to unusual rules, would enhance benchmarking effectiveness and prevent LLMs from becoming familiar with the games. The evaluation of Large Language Models (LLMs) faces significant challenges, such as Benchmark Data Contamination (BDC) [36] and dataset leakage [37], which undermine the reliability and fairness of performance assessments. These issues arise from the exposure of models to benchmark-related data during training, leading to skewed results and compromised generalization.

Custom game-based benchmarks, that has no publicly available rules or game plays, offer a promising alternative by shifting away from static datasets and providing dynamic, adaptive scenarios that reduce the risk of contamination and leakage. This approach not only enhances evaluation robustness but also fosters innovation in assessing LLM capabilities in diverse and interactive contexts.

The simplicity of the games used in this benchmark facilitates basic evaluation but may not challenge LLMs' strategic capabilities as much as more complex games like chess or Go. The fact that current LLMs have not mastered even these simple games provides valuable insights into their capabilities and limitations. Expanding the evaluation to larger grids, such as 4×4 or 5×5 for Tic-Tac-Toe or 19×19 for Gomoku, could present additional challenges and provide a clearer indicator of LLM performance.

Relying on predefined prompts to guide LLMs' moves may not fully capture their potential for independent strategic thinking or their ability to respond to changing game states. Although we updated prompts dynamically to warn LLMs of invalid moves, further techniques, such as providing all previous invalid moves, could be explored to reduce invalid move numbers and disqualifications. Additionally, alternative output extraction methods, such as function calling, could be explored.

While the prompts used in this study were sent in isolation, certain critical prior information, such as invalid moves, was included to enhance the LLMs' ability to avoid repeating errors. Incorporating the full game history into each prompt was avoided due to limitations in LLM context windows and concerns over computational costs associated with per-token usage. Moreover, the decision to exclude full histories aligns with the study's objective of analyzing the LLMs' ability to make strategic decisions based on a limited, structured understanding of the game state rather than relying solely on extensive context memory. Future work could explore how varying levels of historical context affect performance. Additionally, given the constraints of per-token costs, we chose to perform 5 rounds per combination, but future studies should aim for more rounds to improve the robustness and reliability of the results.

This study tested LLMs using structured prompts. Future research should investigate how these prompts influence LLM performance and how variations in prompt structure might affect their understanding of game states and subsequent moves. Such insights could help optimize LLMs for more complex and varied applications.

Additionally, a more detailed analysis could be conducted on how the LLMs' performance is influenced by specific prompt components, such as the effect of telling the model that it is an "adept strategic player" versus a "regular player." Testing LLMs under these different prompts could help assess whether LLMs are truly engaging in strategic thinking or merely mimicking known moves. This would also offer insights into how prompt engineering influences LLM behavior, providing a foundation for improving their decision-making in more complex and varied contexts. While we were unable to pursue this comparison within the scope of this study, it is an important direction for future research.

The evaluation metrics used in this study revealed a wide range of LLM capabilities. While these metrics provide a good indication of performance, they may not fully capture the strategic complexity of the models. Further analysis of the moves, drawn from the JSON and PNG files shared in the GitHub repository, could offer a more detailed assessment of game progress over time. For example, evaluating the creation of winning opportunities, such as aligning moves in Tic-Tac-Toe, Connect Four, and Gomoku, can provide insights into proactive strategic thinking by the LLMs. Contributions to the repository with suggestions and implementations of new metrics and methods are encouraged. Focusing on a select group of LLMs might not capture the full diversity of strategic approaches across available models, highlighting the importance of including a broader array in future research. The rapidly expanding landscape of LLMs, with new models and improved versions emerging frequently, necessitates continuous updates to benchmarks. We welcome submissions of other and new LLMs using the open-source game simulation software.

Future work could explore several promising directions to extend research and deepen our understanding of LLM capabilities in strategic games and beyond. Multi-agent collab- oration scenarios, where multiple LLMs work together against a common opponent or compete in teams, could assess their abilities in coordination, cooperation, and competitive strategy. Comparing newer versions of LLMs against those tested in this study could track progress and improvements in AI strategic gaming capabilities over time.

This study suggests several avenues for future research and development. Firstly, improving LLMs' abilities to interpret and act on visual data is crucial, as evidenced by high invalid move rates in illustration and image prompts. Enhancing visual processing capabilities could significantly boost overall performance and utility. Secondly, further research is needed to enhance LLMs' decision-making processes in more complex environments.

7. CONCLUSION

This study introduces a novel and extensible benchmark for LLMs using grid-based games like Tic-Tac-Toe, Connect Four, and Gomoku. The open-source game simulation code, avail- able on GitHub, enables LLMs to compete and generates data files in JSON, CSV, TXT, and PNG formats for leaderboard rankings and further analysis. We present the results from games among leading LLMs, incorporating Claude 3.5 Sonnet and Claude 3 Sonnet by Anthropic, Gemini 1.5 Pro and Gemini 1.5 Flash by Google, GPT-4 Turbo and GPT-4o by OpenAI, and Llama3-70B by Meta, and encourage submissions from other LLMs.

Analyzing the performance of these models over 2,310 games revealed significant variations in their capabilities, particularly their struggles with complex and visually based prompt formats. Comparisons with a random play generator underscore the LLMs' superior yet still developing capacity for strategic decision-making. LLMs perform relatively well with simpler formats, such as list prompts for Tic-Tac-Toe and Connect Four, but their performance declines with more complex prompts, especially those involving illustrations and images. This trend indicates current limitations in LLMs' ability to interpret and act on visual data and manage increased game complexity. Additionally, the models tend to make more invalid moves with complex prompts, highlighting the need for improved strategic decision-making processes.

Future research could expand the types and complexity of games, test more sophisticated prompt engineering techniques, and explore the effects of different prompt structures. Advancements in LLMs' strategic thinking and decision-making could enhance applications in robotics, autonomous systems, and interactive AI. The modular, open-source benchmarking framework encourages community contributions, enabling a more dynamic and comprehensive evaluation of LLM capabilities.

In conclusion, while the current evaluation highlights both the strengths and limitations of LLMs, it also points to the need for ongoing research to enhance their ability to process complex and visual data, improve decision-making processes, and develop more sophisticated benchmarking tools. This continuous development will ultimately broaden the applicability and effectiveness of LLMs in various real-world tasks.

ACKNOWLEDGMENTS

This study was partially supported by Florida Polytechnic University with grant number GR-24SUMR-OT

REFERENCES

[1] H. Naveed, A. U. Khan, S. Qiu, M. Saqib, S. Anwar, M. Usman, N. Akhtar, N. Barnes, and A. Mian. A comprehensive overview of large language models. arXiv, 2023.

[2] B. Goertzel and C. Pennachin, editors. Artificial General Intelligence, volume 2. Springer, New York, NY, USA, 2007.

[3] I. Sutskever. The exciting, perilous journey toward agi, 2024. Available online: https://tedai-sanfrancisco.ted.com/speakers/ilya-sutskever/ the exciting perilous journey toward agi (accessed on 7 Dec 2024).

[4] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. Attention is all you need. In Advances in Neural Information Processing Systems, volume 30, 2017.

[5] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv, 2018. arXiv:1810.04805.

[6] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever. Improving language understanding by generative pre-training, 2024. Available online: https://paperswithcode.com/paper/improving-language-understanding-by (accessed on 7 Dec 2024).

[7] S. Bubeck, V. Chandrasekaran, R. Eldan, J. Gehrke, E. Horvitz, E. Kamar, P. Lee, Y. T. Lee, Y. Li, S. Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2303.12712 (accessed on 7 Dec 2024).

[8] G. Team, R. Anil, S. Borgeaud, Y. Wu, J. B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, et al. Gemini: A family of highly capable multimodal models. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2312.11805 (accessed on 7 Dec 2024).

[9] Anthropic. Model card and evaluations for claude models, 2024. Available online: https://www-files.anthropic.com/production/images/Model-Card-Claude-2.pdf (accessed on 7 Dec 2024).

[10] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozie`re, N. Goyal, E. Hambro, F. Azhar, et al. Llama: Open and efficient foundation language models. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2302.13971 (accessed on 7 Dec 2024).

[11] Kaddour, J. Harris, M. Mozes, H. Bradley, R. Raileanu, and R. McHardy. Challenges and applications of large language models. arXiv, 2023. arXiv:2307.10169.

[12] S. Minaee, T. Mikolov, N. Nikzad, M. Chenaghlu, R. Socher, X. Amatriain, and J. Gao. Large language models: A survey. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2402.06196 (accessed on 7 Dec 2024).

[13] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, et al. A survey on evaluation of large language models. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2307.03109 (accessed on 7 Dec 2024).

[14] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/ abs/1804.07461 (accessed on 7 Dec 2024).

[15] A. Wang, Y. Pruksachatkun, N. Nangia, A. Singh, J. Michael, F. Hill, O. Levy, and S. R. Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/1905.00537 (accessed on 7 Dec 2024).

[16] P. Liang, R. Bommasani, T. Lee, D. Tsipras, D. Soylu, M. Yasunaga, Y. Zhang, D. Narayanan, Y. Wu, A. Kumar, et al. Holistic evaluation of language models. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2211.09110 (accessed on 7 Dec 2024).

[17] Hendrycks, C. Burns, S. Basart, A. Zou, M. Mazeika, D. Song, and J. Steinhardt. Measuring massive multitask language understanding. CoRR, abs/2009.03300, 2020. Available online: https://arxiv.org/abs/2009.03300 (accessed on 7 Dec 2024).

[18] A. Srivastava, A. Rastogi, A. Rao, A. A. M. Shoeb, A. Abid, A. Fisch, A. R. Brown, A. Santoro, A. Gupta, A. Garriga-Alonso, et al. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2206.04615 (accessed on 7 Dec 2024).

[19] P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. arXiv, 2018. arXiv:1803.05457.

[20] S. Lin, J. Hilton, and O. Evans. Truthfulqa: Measuring how models mimic human falsehoods. arXiv, 2021. arXiv:2109.07958.

[21] R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, and Y. Choi. Hellaswag: Can a machine really finish your sentence? arXiv, 2019. arXiv:1905.07830.

[22] C. White, S. Dooley, M. Roberts, A. Pal, B. Feuer, S. Jain, et al. Livebench: A challenging, contamination-free llm benchmark. arXiv, 2024. arXiv:2406.19314.

[23] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, et al. A survey on evaluation of large language models. ACM Transactions on Intelligent Systems and Technology, 15(3):1–45, 2024.

[24] Z. Guo, R. Jin, C. Liu, Y. Huang, D. Shi, L. Yu, et al. Evaluating large language models: A comprehensive survey. arXiv, 2023. arXiv:2310.19736.

[25] Andrew Ng. We need better evals for llm applications, 2024. Available online: https://www.deeplearning.ai/the-batch/we-need-better-evals-for-llm-applications/ (accessed on 7 Dec 2024).

[26] Q. Tan, A. Kazemi, and R. Mihalcea. Text-based games as a challenging benchmark for large language models, 2024. Available online: https://openreview.net/forum?id=2g4m5SknF (accessed on 7 Dec 2024).

[27] D. Qiao, C. Wu, Y. Liang, J. Li, and N. Duan. Gameeval: Evaluating llms on conversational games. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2308.10032 (accessed on 7 Dec 2024).

[28] Y. Wu, X. Tang, T. M. Mitchell, and Y. Li. Smartplay: A benchmark for llms as intelligent agents. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2310.01557 (accessed on 7 Dec 2024).

[29] E. Akata, L. Schulz, J. Coda-Forno, S. J. Oh, M. Bethge, and E. Schulz. Playing repeated games with large language models. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2305.16867 (accessed on 7 Dec 2024).

[30] C. F. Tsai, X. Zhou, S. S. Liu, J. Li, M. Yu, and H. Mei. Can large language models play text games well? current state-of-the-art and open questions. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2304.02868 (accessed on 7 Dec 2024).

[31] C. Fan, J. Chen, Y. Jin, and H. He. Can large language models serve as rational players in game theory? a systematic analysis. arXiv [Cs.CL], 2024. Available online: http://arxiv.org/abs/2312.05488 (accessed on 7 Dec 2024).

[32] J. Duan, R. Zhang, J. Diffenderfer, B. Kailkhura, L. Sun, E. Stengel-Eskin, et al. Gtbench: Uncovering the strategic reasoning limitations of llms via game-theoretic evaluations. arXiv, 2024. arXiv:2402.12348.

[33] A. Costarelli, M. Allen, R. Hauksson, G. Sodunke, S. Hariharan, C. Cheng, et al. Game-bench: Evaluating strategic reasoning abilities of llm agents. arXiv, 2024. arXiv:2406.06613.

[34] R. Gallotta, G. Todd, M. Zammit, S. Earle, A. Liapis, J. Togelius, and G. N. Yannakakis. Large language models and games: A survey and roadmap. arXiv, 2024. arXiv:2402.18659.

[35] S. Hu, T. Huang, F. Ilhan, S. Tekin, G. Liu, R. Kompella, and L. Liu. A survey on large language model-based game agents. arXiv, 2024. arXiv:2404.02039.

[36] Xu, C., Guan, S., Greene, D., & Kechadi, M. (2024). Benchmark Data Contamination of Large Language Models: A Survey. arXiv preprint arXiv:2406.04244.

[37] Xu, R., Wang, Z., Fan, R. Z., & Liu, P. (2024). Benchmarking benchmark leakage in large language models. arXiv preprint arXiv:2404.18824.

[38] Topsakal, O., & Harper, J. B. (2024). Benchmarking Large Language Model (LLM) Performance for Game Playing via Tic-Tac-Toe. Electronics, 13(8), 1532. https://doi.org/10.3390/electronics13081532

[39] Topsakal, O., Edell, C. J., & Harper, J. B. (2024). Evaluating Large Language Models with Grid-Based Game Competitions: An Extensible LLM Benchmark and Leaderboard. arXiv [Cs.AI]. Available online: http://arxiv.org/abs/2407.07796 (accessed on 7 Dec 2024).

[40] LLM Game Benchmark. 2024. Available online: https://github.com/research-outcome/LLM-Game-Benchmark/ (accessed on 7 Dec 2024).

[41] L. V. Allis, H. J. van den Herik, and M. P. Huntjens. Gomoku solved by new search techniques. Computational Intelligence, 12(1):7–72, 1996.

[42] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, et al. A survey of large language models. arXiv, 2023. arXiv:2303.18223.

BIOGRAPHIES

Oguzhan TOPSAKAL earned his BS in Computer Engineering from Istanbul Technical University and his MS and Ph.D. in Computer Science from the University of Florida. He served as a faculty member at Florida Polytechnic University (2018–2024), leading research on digitizing facial plastic surgeries and deep learning in medicine. In Fall 2024, he joined the University of South Florida, bringing over 15 years of experience in software development, teaching, and research. His expertise includes deep learning, machine learning, software engineering, and augmented reality, with current research focusing on AI applications in medicine and education.

Colby J. EDELL is currently pursuing a B.S. in Computer Science at Florida Polytechnic University in Lakeland, FL, USA. During the summer of 2024, he collaborated with Dr. Oguzhan Topsakal to develop an extensible, web-based benchmark for evaluating large language models in game-playing tasks. Additionally, he worked with Dr. Ayesha S. Dina on research in intrusion detection for vehicular networks, aiming to advance the development of more secure and intelligent systems in this field.

Jackson B. HARPER is pursuing a B.S. in Computer Science at Florida Polytechnic University in Lakeland, FL, USA. During the spring and summer of 2024, he collaborated with Dr. Oguzhan Topsakal to evaluate large language models using a mobile app and contributed to the development of an extensible, web-based benchmark for assessing LLMs in game-playing tasks. Additionally, he gained valuable hands-on experience as a network team intern, showcasing a strong commitment to innovation, problem-solving, and advancing technology through both research and practical applications.