SportsMetrics: Blending Text and Numerical Data to Understand Information Fusion in LLMs

Anonymous ACL submission

Abstract

Large language models hold significant potential for integrating various data types, such as text documents and database records, for advanced analytics. However, blending text and numerical data presents substantial challenges. LLMs need to process and cross-reference entities and numbers, handle data inconsistencies and redundancies, and develop planning capabilities such as building a working memory for managing complex data queries. In this paper, we introduce four novel tasks centered around sports data analytics to evaluate the numerical reasoning and information fusion capabilities of LLMs. These tasks involve providing LLMs with detailed, play-by-play sports game descriptions, then challenging them with adversarial scenarios such as new game rules, longer durations, scrambled narratives, and analyzing key statistics in game summaries. We conduct extensive experiments on NBA and NFL games to assess the performance of LLMs on these tasks. Our benchmark, SportsMetrics, introduces a new mechanism for assessing LLMs' numerical reasoning and fusion skills.

1 Introduction

Large language models (LLMs) are more powerful than ever. OpenAI's GPT-4 Turbo (2023) features a 128k context window, allowing it to process over 300 pages of text in a single prompt. Claude v2.1 (2023) steps it up with a 200k token window, equivalent to roughly 150,000 words or more than 500 pages. Mistral AI (2023) has created a sparse mixture of experts model capable of processing up to 32k tokens. The developments suggest language models can now engage with vast amounts of text content and data, opening doors to exciting new applications in various domains.

One of the most promising uses of LLMs is in handling a combination of unstructured texts and structured data. For example, determining if a patient can be discharged from the hospital may involve reviewing doctor notes, radiology and pathology reports, lab results, and other records that blend

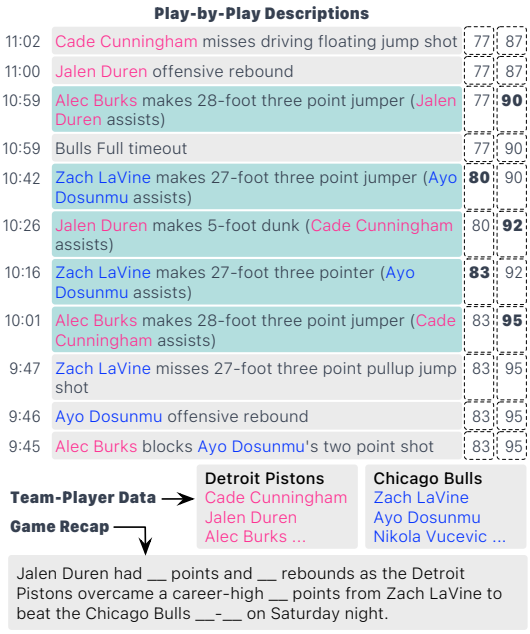

Figure 1: Play-by-plays of an NBA game. We include timestamps, player actions, team affiliations and a game recap. Total points for both teams are indicated in dotted circles and are withheld from LLMs.

text and structured data (Adams et al., 2021; Bardhan et al., 2022; Veen et al., 2023; Ben Abacha et al., 2023); LLM Assistants for online shopping need to process product catalogs, sales transactions, and customer queries (Brynjolfsson et al., 2023; Loten, 2023). However, summarizing key details from a mix of unstructured and structured sources remains a considerable challenge. An LLM must navigate text descriptions, link entities, aggregate numbers, handle discrepancies, and beyond.

Information fusion focuses on synthesizing information from multiple textual sources to derive meaningful conclusions (Barzilay et al., 1999). Current approaches involve summarizing multiple text documents, providing concise answers to user queries, and integrating summarization with natural language inference to deduce information (Bhaskar et al., 2023; Caciularu et al., 2023; Sprague et al., 2022; Bostrom et al., 2022). The output is often a short text summary, the quality of which is difficult

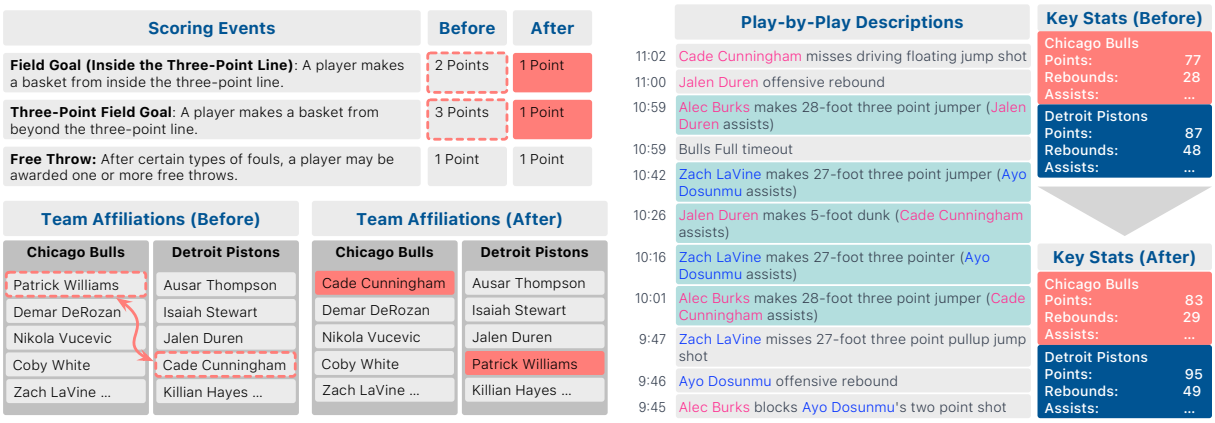

Figure 2: (TOP LEFT) We examine the impact of changing game rules on final scores. For basketball, scoring events such as free throws, three-pointers, field goals, vary from 1 to 3 points. We ask LLMs to maintain these scoring events but under a new rule where each is worth only 1 point. (BOTTOM LEFT) We randomly swapped player team affiliations in the table without altering the game's play-by-play records. (RIGHT) LLMs are provided with detailed play-by-play descriptions of a sports game and player team affiliations. Their job is to use this information to update key game statistics in a JSON format.

to evaluate (Deutsch et al., 2021). Our approach differs by focusing on the numerical aspect of information fusion (Geva et al., 2020). We enable the LLM to navigate through lengthy texts, gather crucial statistics, and develop a working memory to manage complex data queries.

We introduce `SportsMetrics`, a benchmark designed to assess LLMs' abilities in numerical reasoning and data fusion. This benchmark provides LLMs with detailed, play-by-play descriptions of sports games, including timestamps, player actions, and team affiliations, as illustrated in Figure 1. We focus on four novel tasks to evaluate LLMs in adversarial scenarios: (a) *adapting to new game rules*, (b) *handling lengthy game descriptions*, (c) *managing scrambled game narratives*, and (d) *analyzing critical statistics in game summaries*. E.g., an LLM might be asked to complete a basketball game recap by inserting missing key statistics, which requires the development of a working memory for game stats and reasoning skills.

Our `SportsMetrics` benchmark presents three main benefits. First, it uses a broad range of sports data; they are dynamic narratives that LLMs cannot easily memorize. Second, it allows us to evaluate LLMs' ability to track key statistics such as team points, assists, blocks, steals, and more, while also offering an overall game efficiency score for direct LLM comparison. Lastly, its use of widely understood sports terminology makes it more accessible to researchers than specialized medical language, making it an ideal benchmarking tool. While our current focus is on English, `SportsMetrics` also holds promise for multilingual applications.

2 Related Work

There is a growing need for a benchmark to evaluate LLMs' information fusion capabilities, which offers clear, quantitative scores for comparing various LLMs. For example, Chatbot Arena (Zheng et al., 2023) utilizes Elo ratings (Boubdir et al., 2023), MT-Bench comprises of 80 multi-turn questions, and MMLU focuses on a model's multitask accuracy across 57 tasks (Hendrycks et al., 2021). Multi-document summarization offers a promising benchmark (Huang et al., 2021; Wang et al., 2022; Xu et al., 2023). However, developing a summary scoring system poses challenges due to variables such as summary length, content coverage, and faithfulness (Cao et al., 2022; Liu et al., 2023c; Krishna et al., 2023). Furthermore, we need to ensure that benchmark data has not been part of LLM pretraining datasets (Liu et al., 2023c; Li et al., 2023). Sports game data, which combines static knowledge with player dynamics, presents an untapped opportunity for benchmarking LLMs.

Combining information from a blend of textual and numerical records poses a significant challenge. In traditional multi-document summarization, the system creates a concise summary from a set of topically related documents. Giorgi et al. (Giorgi et al., 2023) show that this task remains difficult in an "open-domain" setting, where the document set is generated by a retriever and may include irrelevant information. With the growing popularity of retrieval-augmented generation (RAG) (Karpukhin et al., 2020; Liu et al., 2022), there is an increasing need to accurately fuse information from various sources. We explore information fusion by examin-

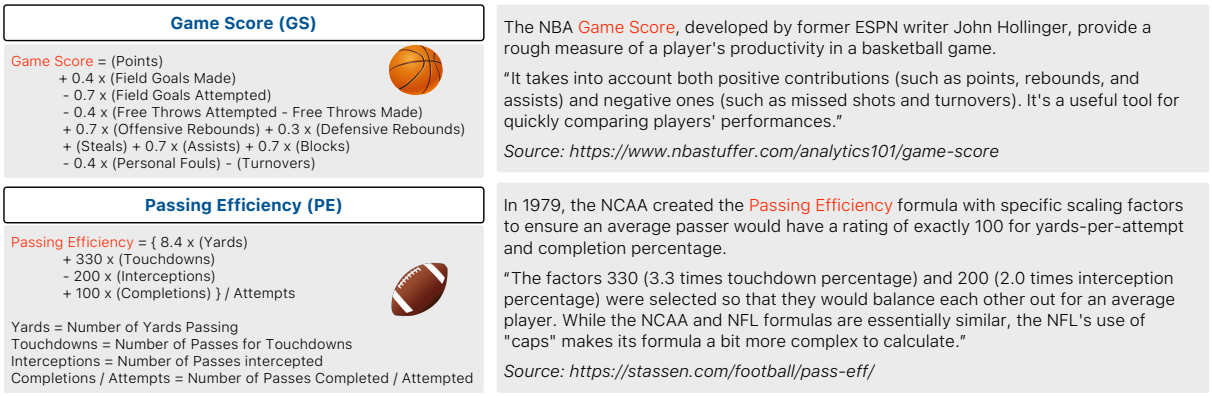

Figure 3: We adopt the NBA's *Game Score*, originally designed for player evaluation, to measure a team's overall efficiency. For American football, we apply NCAA's *Passing Efficiency* formula.

ing how LLMs cross-reference players and actions, and aggregate data across play-by-play descriptions to compile key game statistics.

Our work relates to numerical reasoning, which uses arithmetic reasoning to tackle mathematical word problems. Prior datasets in this area include MathQA (Amini et al., 2019), GSM8k (Cobbe et al., 2021), SVAMP (Patel et al., 2021), and MATH (Liu et al., 2023d), targeting grade school level problems, and allowing models to generate answer explanations. The problems typically have brief descriptions, with the challenge lying in creating an expression tree and applying arithmetic knowledge. In contrast, our approach focuses on assessing LLMs' ability to track key statistics across extremely long contexts.

Sports data has been utilized in various natural language tasks, including data-to-text generation for sports games (Lareau et al., 2011; Zhang et al., 2016; Wiseman et al., 2017; van der Lee et al., 2017; Puduppully et al., 2019), real-time game summarization from live commentaries (Edouard et al., 2017; Huang et al., 2020); and other aspects such as sports commentator bias (Merullo et al., 2019). Beyond sports, there's significant interest in annotating and analyzing large-scale game-related corpora, such as reviews and gameplay logs, and summarizing gameplay commentaries (Lukin, 2020; Kicikoglu et al., 2020; Gu et al., 2022; Furman et al., 2022). We anticipate that insights from our `SportsMetrics` benchmark will benefit these areas, enhancing our understanding of game narratives and player dynamics.

3 The `SportsMetrics` Benchmark

We collect NBA and NFL play-by-play data from ESPN.com. The descriptions capture the essence of each game. They are typically written by ESPN's sports journalists, who are experts in their respective sports. We reached out to ESPN as necessary to ensure adherence to their data policies. In Figure 1, we use "*time*" to indicate the exact moment of each action on the game clock, while "*play*" details the actions occurring at those times. Scoring actions, which change the game's score, are identified but not disclosed to LLMs during our experiments, as are team points. Additionally, we collect data on players' team affiliations and the game's box scores for our analysis.

Our task requires LLMs to track key stats across thousands of play-by-play records, which is a non-trivial effort. An ideal LLM needs to associate each action with the right player and their team in order to calculate team-level statistics. It must also monitor multiple key statistics simultaneously, such as field goals, free throws, rebounds, assists, blocks, steals, personal fouls, and turnovers in a basketball game. We believe an LLM's ability to summarize key details and fill in the missing statistics in game summaries demonstrates its capabilities in data fusion and numerical reasoning.

We need a comprehensive scoring metric to evaluate LLMs' ability to monitor key game statistics. Simply reporting individual metrics such as team points, rebounds, assists, and blocks for each team is inefficient and does not provide a holistic view of game analysis. To address this, we employ expert-developed team statistics formulas, as illustrated in Figure 3. We adopt the NBA's "***Game Score***" by John Hollinger, originally for player evaluation, to measure a team's overall effectiveness in basketball. It considers both positive (points, rebounds, assists) and negative (missed shots, turnovers) factors. For American football, we apply NCAA's "***Passing Efficiency***" formula, as the NFL rule is more complex. In the following sections, we evaluate LLMs under different adversarial scenarios to assess their robustness.

Figure 4: An LLM fills in missing key statistics in game summaries through a three-step process. Initially, the LLM creates an internal JSON object as its memory. It then enriches this memory by adding necessary game or player statistics, where all values are set to null, and further reflects on whether this memory is sufficient to accomplish the task. Lastly, the LLM uses detailed play-by-play and team-player data to update the JSON object's values; it finally utilizes this updated memory to fill in the blanks in the game summary.

3.1 Long-Form Game Narratives

We begin by examining LLMs' ability to reason over long contexts. For example, Liu et al. (2023b) introduced two tasks, multi-document QA and key-value retrieval, which require the model to identify relevant information within long contexts. They found that LLMs' performance significantly deteriorates when they have to access relevant information in the middle of long contexts. Our study goes a step further, requiring LLMs to not only identify relevant actions but also accurately track statistics throughout long-form game narratives.

In this task, each LLM is provided detailed play-by-play descriptions of a sports game, including timestamps and specific actions. The players' team affiliations are listed in two rows, representing each team. The LLM's task is to use the play-by-plays to update key game statistics within a JSON object, initially filled with null values. For long-context LLMs such as GPT-4 Turbo, Claude 2.1, and Gemini Pro (Anil et al., 2023), we provide the entire game's data at once for processing. For LLMs with 4k or 8k tokens context, we break the game down into four quarters. The LLM gathers statistics quarter by quarter. It generates a JSON object that holds values from each quarter. These are then added up to derive game-level statistics.

We use comprehensive, expert-devised formulas to evaluate LLMs in tracking game statistics. For NBA games, we monitor 11 key statistics: *team points*, *field goals made*, *field goals attempted*, *free throws made*, *free throws attempted*, *offensive rebounds*, *defensive rebounds*, *steals*, *assists*, *blocks*, and *personal fouls*.[1] Moreover, we calculate 'Game Score' to measure a team's overall effectiveness in basketball. For NFL games, we track *passing yards*, *touchdowns*, *interceptions*, and *pass completions* and *attempts*. These additional stats allow for the computation of 'Passing Efficiency.'

3.2 The Impact of Changing Game Rules

It is important to understand LLMs' ability to make decisions under changing world rules. LLMs possess extensive knowledge from pretraining on the Internet, books, and other texts. This knowledge, held in their parametric memory, might not always align with the external evidence given to the model. Therefore, LLMs need to adjust to changing rules. Xie et al. (2023) highlight the importance of knowing when to trust a model's own knowledge. Meng et al. (2023) explored finetuning LLMs to alter specific knowledge, but such changes are often irreversible. Here, we propose two tasks to evaluate LLMs' abilities in adapting to new game rules.

[1]We exclude *turnovers* from tracking due to limitations in the data. Play-by-play descriptions may not capture every turnover, making it difficult for the model to track them accurately. When necessary, we rely on the ground-truth Turnover count from the box score to calculate the Game Score.

Model	Release Date	Context Len	Input	Output	Organization
Claude-2.1	11.21.2023	200,000	$.008	$.024	Anthropic
GPT-4-1106-preview	11.06.2023	128,000	$.01	$.03	OpenAI
Gemini-Pro	12.06.2023	32,000	$.001	$.002	Deepmind
GPT-3.5-Turbo-1106	11.06.2023	16,385	$.001	$.002	OpenAI
Mistral-7B-Instruct-v0.1	09.27.2023	8,000	—	—	Mistral
GPT-3.5-Turbo-0613	06.13.2023	4,096	$.0015	$.0015	OpenAI
Llama-2-13B-Chat	07.18.2023	4,096	—	—	Meta

Table 1: LLMs used in this study. Prices are per 1,000 tokens. Llama-2 and Mistral-7B are free and open-source.

New Scoring Rules We examine the impact of changing game rules on final scores. For basketball, scoring events such as free throws, three-pointers, field goals, vary from 1 to 3 points. We ask LLMs to maintain these scoring events but under a new rule where each action is worth only 1 point. This contradicts LLMs' existing knowledge, challenging them to recalibrate game scores accordingly. Ground-truth scores under this rule are obtained by counting the total number of scoring actions to determine each team's total points.

Player Swapping We randomly swapped player team affiliations in the table without changing the game's play-by-play records, as illustrated in Figure 2. Ground-truth team scores for this task are calculated by summing individual player scores under their new affiliations. This task allows us to vary the degree of conflict between the model's existing knowledge and the provided evidence. Swapping more players increases the task's difficulty.

3.3 Robustness Against Noise

Shuffling Play-by-Plays We present an adversarial challenge where we shuffle basketball game play-by-play descriptions and then ask LLMs to track the total points of each team. We choose basketball because adjacent actions in this context do not show strong causal relationships. Changing the sequence of scoring actions does not affect the teams' total points. We anticipate that long-context LLMs will produce consistent or similar final game scores. To avoid confusing the model, we maintain the original order of timestamps.

We can also adjust the frequency of scoring plays in a game, making it more or less challenging for LLMs to process the narrative. By choosing a probability p from a set of values {-50%, -20%, 0, +20%, +50%}, we can either duplicate non-scoring plays (thereby decreasing scoring play density and extending the game narrative) or remove them (increasing scoring play density). Further, to test the LLM's inherent knowledge, we randomly select players from each team in NFL games and assign them new names, such as characters from science fictions. This approach evaluates the model's ability to adapt to changes in player identities. These alterations do not introduce new players or change the total points scored in the game; it simply varies the narrative's complexity.

Figure 5: Effective working memory is key in this task. While a sophisticated structure is possible, it also increases the likelihood of errors when populating values.

3.4 Planning for Complex Data Queries

In this task, LLMs fill in missing key statistics from game summaries (e.g., from ESPN). The process unfolds in three steps, illustrated in Figure 4. First, the LLM creates an internal JSON object memory, initially with placeholders for team points. Next, it enriches this memory by adding crucial game or player statistics. During a self-reflection phase, the LLM evaluates if its JSON memory can accurately complete the missing statistics for the given game recap. If it can, it responds with this memory; if not, it further refines the memory structure. Finally, using the detailed play-by-play and team-player data, the LLM updates the key statistics in the JSON format, then uses this information to fill in the blanks in the game summary. Figure 5 illustrates various LLM attempts building a memory.

Our task is inspired by several studies on LLM planning. Unlike LLM+P which uses the Planning Domain Definition Language (PDDL) for problem-

	System	ΔGScore	ΔPoints	ΔNewRule	ΔSwap	ΔShuffle
Long-Context	GPT-3.5-Turbo-1106	33.50	**9.45**	**14.10**	**13.53**	**9.89**
(16k+ Tokens)	Gemini-Pro	**32.30**	17.62	25.99	17.78	14.85
	GPT-4-1106-preview	51.97	25.17	14.55	39.91	49.57
	Claude-2.1	55.16	21.73	22.28	17.12	31.11
Standard	GPT-3.5-Turbo-0613	114.28	94.34	18.22	88.25	89.11
(4k to 8k Tokens)	Mistral-7B-Instuct	123.49	73.53	26.79	70.69	103.24
	Llama-2-13B-Chat	110.69	70.77	83.04	53.98	81.09

Table 2: Average absolute difference between model predictions and the actual scores on NBA data for tracking a team's total points (**Points**) and all key game statistics (**GScore**). Moreover, we evaluate LLMs' performance in three adversarial scenarios: Δ**NewRule**, Δ**Swap** and Δ**Shuffle**.

solving (Liu et al., 2023a), we simplify the process by requiring only a valid JSON object for working memory. Relevant studies such as Reflexion (Shinn et al., 2023), ReAct (Yao et al., 2023b), and Tree-of-thought (Yao et al., 2023a) have also influenced our approach. Sumers et al. (2023) have developed a framework for integrating planning into LLM agents. Prior studies have focused on ALFWorld's interactive TextWorld environments. Our method are focused on sports, which involves masking key statistics in game recaps by sports journalists, then converting them into task data points for LLMs. We assess LLMs by their accuracy in filling in missing key statistics from game summaries.

4 Experiments

We evaluate various LLMs in our `SportsMetrics` benchmark. These models are listed in Table 1 and split into two categories: long-context LLMs, capable of processing over 16k tokens, and standard LLMs, handling 4k to 8k tokens. Our evaluation focuses on their ability to accurately track a team's total points (**Points**) and all key game statistics (**GameScore**). We measure the average absolute difference (deviation) between the models' predictions and the actual box scores, denoted as ΔPoints and ΔGScore, respectively.[2]

Our dataset comprises 28,492 NBA games and 5,867 NFL games spanning two decades from 2002 to 2023, available through ESPN's archives. We randomly selected 100 games from each sport for our test set. On average, NBA games contain 466 plays and NFL games 173 plays. An average NBA game includes 6,229 tokens, while an NFL game has 6,166 tokens, with maximum lengths reaching 7,322 and 7,659 tokens, respectively.

LLMs' ability to integrate information is tested under three adversarial scenarios: (a) '**NewRule**,' which assigns every scoring action just one point, regardless of the move, (b) '**Swap**' which randomly selects two players from each team to swap their affiliations in the team-player table, (c) '**Shuffle**,' which duplicates any non-scoring action with a 20% chance (p=0.2) before shuffling the play-by-plays. We assess LLMs' performance in these scenarios and report the deviation of predicted team points from actual scores as ΔNewRule, ΔSwap and ΔShuffle.

In Table 2, we present our findings from the NBA section of our dataset. With Δ representing the gap between predictions and actual scores, smaller values are preferable. We find that long-context LLMs significantly outperform standard LLMs across all tasks. GPT-3.5-Turbo-1106 leads in performance in every task except for ΔGScore, where Gemini-Pro has a slight edge. Long-context models have been released recently in late 2023. These results demonstrate their remarkable ability in identifying relevant actions from game play-by-plays, attributing each action to the right player and team, and aggregating numerical data to compute final team points and GameScore. This requires a level of numerical reasoning that humans are adept at but it is still new territory for LLMs.

In Figure 6, we organize games based on the length (number of tokens) of their play-by-play descriptions, with the x-axis showing the games and the y-axis the deviation scores from various LLMs, where lower scores indicate better performance. We perform a regression analysis to demonstrate each LLM's trend in handling games of increasing length. GPT-3.5-Turbo-1106 and Gemini-Pro stand out, maintaining nearly flat curves, which cor-

[2] ΔGScore consistently shows higher values compared to ΔPoints because it goes beyond counting a team's points. It offers a full game analysis by requiring the LLM to consolidate key statistics such as points, rebounds, steals, assists and more into an overall score. Considering only team points is insufficient, especially in sports like soccer where scoring is rare. When necessary, we can convert GameScore to points by zeroing out other stats.

	System	ΔYards	ΔATT	ΔCOMP	ΔTD	ΔINT	ΔPE
Long-Context	GPT-4-1106-Preview	**34.77**	**4.44**	**2.96**	**0.17**	**0.13**	**14.33**
(16k+ Tokens)	Claude-2.1	52.53	5.43	3.75	0.29	0.22	17.53
	GPT-3.5-Turbo-1106	64.87	7.80	4.73	0.49	0.30	18.43
	Gemini-Pro	85.14	12.68	6.87	0.83	0.52	26.17
Standard	GPT-3.5-0613	105.68	24.11	15.80	1.09	0.60	**89.56**
(4k to 8k Tokens)	Llama-2-13B-Chat	244.48	22.37	19.66	1.47	1.03	191.76
	Mistral-7B-Instuct	119.31	17.64	9.05	1.23	0.69	202.07

Table 3: Discrepancies between model predictions and actual scores on NFL stats, including yards (**Yards**), attempts (**ATT**), completions (**COMP**), touchdowns (**TD**), interceptions (**INT**) and passing efficiency (**PE**).

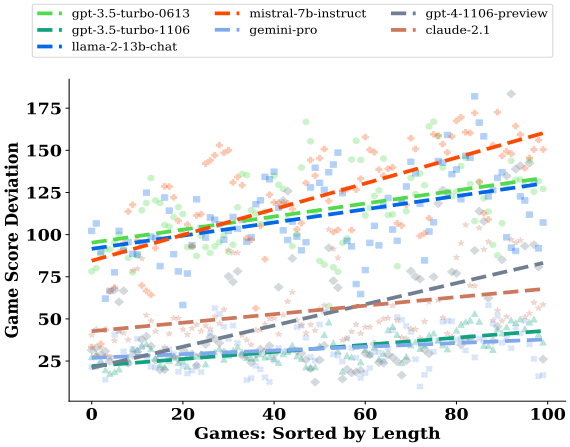

Figure 6: We organize games based on the length of their play-by-plays, with the x-axis showing the games and the y-axis the deviation scores; lower scores indicate better performance. GPT-3.5-Turbo-1106 and Gemini-Pro stand out here, maintaining nearly flat curves.

responds with their superior performance as shown in Table 2. By contrast, GPT-4-1106-Preview does well in shorter games but face difficulties in aggregating key statistics for longer games. Additionally, 79% its returned JSON objects contain zeros or null values, contributing to its unsatisfying performance on this task.

We note that basketball teams typically score between 100 to 120 points. Our findings show that the smallest prediction gap for ΔPoints is 9.45, while the largest can exceed 100. This indicates the difficulty in accurately tracking key game statistics over long contexts, as *standard LLMs can produce predictions significantly off from actual scores due to hallucinations.* Among the three adversarial scenarios, the New Rule is relatively simpler as it requires LLMs to assign one point to every scoring action, focusing on counting these actions instead of distinguishing between types (3-pointers vs. free throws) and adding them up for a team's score. In this scenario, Llama-2-13B-Chat scores lower than all other LLMs.

In Table 3, we present NFL data findings. American football's play-by-plays have demonstrated a sequential nature, we cannot apply tests like New Rule, Swap, or Shuffle as with basketball games. Instead, we measure how model predictions deviate from actual scores on key game statistics, including yards (Δ**Yards**), attempts (Δ**ATT**), completions (Δ**COMP**), touchdowns (Δ**TD**), and interceptions (Δ**INT**). We also combine them into Passing Efficiency (Δ**PE**) for a holistic game analysis. Our results suggest that long-context LLMs greatly surpass standard models, with GPT-4-1106-Preview taking the lead, followed by Claude-2.1 and GPT-3.5-Turbo-1106.

Particularly, passing yards are vital in the NFL games, often leading to scoring opportunities like touchdowns and field goals. On average, NFL teams average 200 to 250 passing yards per game. We find that the top model, GPT-4-1106-Preview, exhibits a 34.77-yard discrepancy in passing yards prediction, while the open-source Llama-2-13B-Chat lags significantly in comparison. This highlights the difficulty of tracking passing yards, a task even more challenging than summarizing basketball points, with most models struggling to accurately aggregate such data.

In Figure 7, we test LLMs' robustness against adversarial conditions. In the left subfigure, we vary the difficulty of identifying scoring events by either dropping or duplicating non-scoring events. E.g., at probability p=-0.5, we eliminate any non-scoring event with a 50% chance; at p=0.2, we duplicate any non-scoring event with a 20% chance, before shuffling the entire game description. The y-axis measures the deviation from the actual box score, with smaller values indicating better model performance. We observe that *GPT-3.5-Turbo-1106 and Gemini-Pro perform the best, whose curves are quite flat, indicating their robustness to a varying level of noise in the play-by-plays.* Overall, LLMs perform well when non-scoring events are removed,

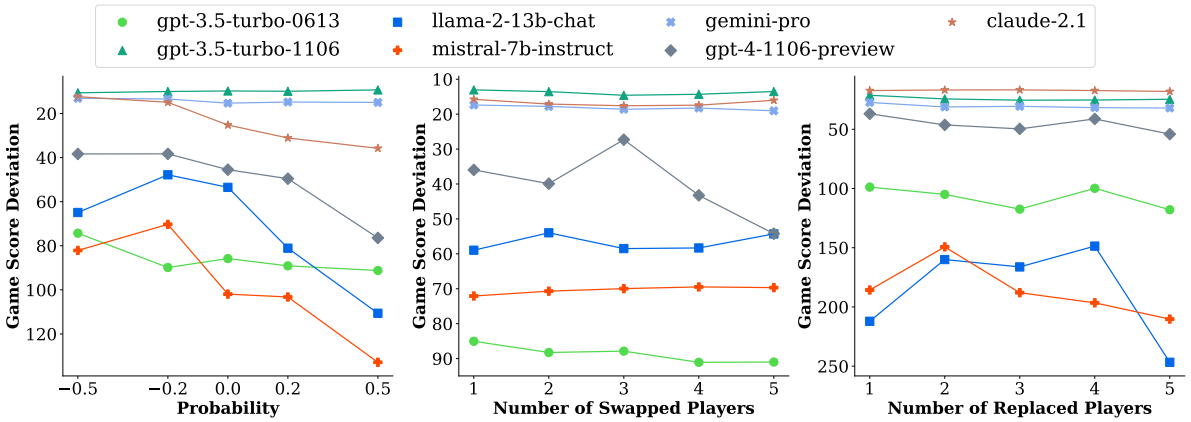

Figure 7: (LEFT) We adjust the difficulty of identifying scoring events by either removing or duplicating non-scoring events. Moreover, we randomly swapped n players' affiliations in the team-player table (MIDDLE) and replaced n players' names with science fiction characters (RIGHT), all without changing the play-by-play texts.

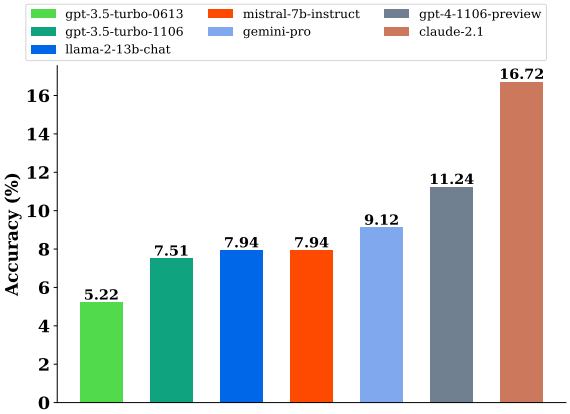

Figure 8: Accuracy of various LLMs in filling missing key statistics from basketball game recaps. Claude-2.1 shows strong performance, while Mistral-7B-Instruct achieves the highest accuracy among standard LLMs.

yet their performance drops as more non-scoring events are added, akin to searching for a needle in a larger haystack.

Further, we randomly swapped n players' affiliations in the team-player table and replaced n players' names with science fiction characters, all without changing the play-by-play texts. Our findings are shown in the middle and right subfigures. We find that Claude-2.1, Gemini-Pro, and GPT-3.5-Turbo-1106 are the top performers. Interestingly, *renaming players significantly decreases all models' performance. This suggests LLMs may use familiar basketball player names from their pretraining to guess team scores, rather than analyzing the actual play-by-plays.* GPT-4-1106-Preview is the least adaptable to these adversarial conditions among the long-context LLMs. We also observe a notable performance disparity exists between open-source and proprietary LLMs.

We assess the accuracy of various LLMs in completing missing key statistics from basketball game recaps. The types of missing data include a player's total points, team scores, assists, rebounds, and other stats. An LLM must understand the recap's context to precisely estimate the missing statistic. To do this, LLMs create a JSON object as its working memory. They then calculate the needed statistics using play-by-play and team-player data and use this memory object to fill in the blanks.

Figure 8 presents the results of this task. Claude-2.1 shows strong performance, while Mistral-7B-Instruct achieves the highest accuracy among standard LLMs. This task requires that LLMs possess strong instruction-following capabilities to build an effective working memory. Figure 5 provides sample working memories from various LLMs. Although complex structures are possible, they increase the risk of errors when populating values. Models such as GPT-4-1106-Preview and Llama-2-13B-Chat face difficulties in creating a working memory. They hallucinate field values or fail to accurately fill fields with aggregated values from play-by-play data. By contrast, Claude-2.1's memory structure is the best in terms of efficiency, focusing on essential game statistics. Our task crucially evaluates LLMs' memory management skills when handling complex data queries.

5 Conclusion

We introduce `SportsMetrics`, a novel benchmark designed to evaluate LLMs in sports data analytics. It assess LLMs' numerical reasoning and fusion abilities through challenges such as new game rules, lengthy descriptions, scrambled narratives and key stats analysis in game summaries. `SportsMetrics` highlights LLMs' potential in fields such as multi-player gaming and collaborative workspaces.

6 Limitations

Our research focuses on NBA and NFL games, which are major sports with rich datasets. We are interested in exploring the generalizability of our findings to other sports. For example, soccer and cricket have distinct play styles and rules, which might challenge LLMs in unique ways. Our study has explored multiple adversarial scenarios, such as new game rules and scrambled game narratives. Such drastic changes might be uncommon in real-world conditions, and the models' ability to handle these scenarios might not translate to improved performance in other analytical tasks. Finally, our scoring system's effectiveness in assessing LLMs' numerical reasoning capabilities in different contexts, such as multiplayer online gaming or collaborative workspaces, remains to be validated. This study explores LLMs' potential in sports analytics. It is important to recognize these limitations when applying our findings to broader contexts.

References

Griffin Adams, Emily Alsentzer, Mert Ketenci, Jason Zucker, and Noémie Elhadad. 2021. What's in a summary? laying the groundwork for advances in hospital-course summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4794–4811, Online. Association for Computational Linguistics.

Aida Amini, Saadia Gabriel, Peter Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. Mathqa: Towards interpretable math word problem solving with operation-based formalisms.

Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Slav Petrov, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, and Emily Pitler. 2023. Gemini: A family of highly capable multimodal models.

Anthropic. 2023. Introducing Claude 2.1. `https://www.anthropic.com/index/claude-2-1`. Accessed on: Nov 21, 2023.

Jayetri Bardhan, Anthony Colas, Kirk Roberts, and Daisy Zhe Wang. 2022. DrugEHRQA: A question answering dataset on structured and unstructured electronic health records for medicine related queries. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1083–1097, Marseille, France. European Language Resources Association.

Regina Barzilay, Kathleen R. McKeown, and Michael Elhadad. 1999. Information fusion in the context of multi-document summarization. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, pages 550–557, College Park, Maryland, USA. Association for Computational Linguistics.

Asma Ben Abacha, Wen-wai Yim, Yadan Fan, and Thomas Lin. 2023. An empirical study of clinical note generation from doctor-patient encounters. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2291–2302, Dubrovnik, Croatia. Association for Computational Linguistics.

Adithya Bhaskar, Alex Fabbri, and Greg Durrett. 2023. Prompted opinion summarization with GPT-3.5. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9282–9300, Toronto, Canada. Association for Computational Linguistics.

Kaj Bostrom, Zayne Sprague, Swarat Chaudhuri, and Greg Durrett. 2022. Natural language deduction through search over statement compositions. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4871–4883, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Meriem Boubdir, Edward Kim, Beyza Ermis, Sara Hooker, and Marzieh Fadaee. 2023. Elo uncovered: Robustness and best practices in language model evaluation.

Erik Brynjolfsson, Danielle Li, and Lindsey Raymond. 2023. Generative ai at work.

Avi Caciularu, Matthew Peters, Jacob Goldberger, Ido Dagan, and Arman Cohan. 2023. Peek across: Improving multi-document modeling via cross-document question-answering. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1970–1989, Toronto, Canada. Association for Computational Linguistics.

Meng Cao, Yue Dong, and Jackie Cheung. 2022. Hallucinated but factual! inspecting the factuality of hallucinations in abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3340–3354, Dublin, Ireland. Association for Computational Linguistics.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems.

Daniel Deutsch, Rotem Dror, and Dan Roth. 2021. A statistical analysis of summarization evaluation metrics using resampling methods. *Transactions of the Association for Computational Linguistics*, 9:1132–1146.

Amosse Edouard, Elena Cabrio, Sara Tonelli, and Nhan Le-Thanh. 2017. You'll never tweet alone: Building sports match timelines from microblog posts. In *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, pages 214–221, Varna, Bulgaria. INCOMA Ltd.

Gregory Furman, Edan Toledo, Jonathan Shock, and Jan Buys. 2022. A sequence modelling approach to question answering in text-based games. In *Proceedings of the 3rd Wordplay: When Language Meets Games Workshop (Wordplay 2022)*, pages 44–58, Seattle, United States. Association for Computational Linguistics.

Mor Geva, Ankit Gupta, and Jonathan Berant. 2020. Injecting numerical reasoning skills into language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 946–958, Online. Association for Computational Linguistics.

John Giorgi, Luca Soldaini, Bo Wang, Gary Bader, Kyle Lo, Lucy Wang, and Arman Cohan. 2023. Open domain multi-document summarization: A comprehensive study of model brittleness under retrieval. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 8177–8199, Singapore. Association for Computational Linguistics.

Yi Gu, Shunyu Yao, Chuang Gan, Josh Tenenbaum, and Mo Yu. 2022. Revisiting the roles of "text" in text games. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6867–6876, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding.

Kuan-Hao Huang, Chen Li, and Kai-Wei Chang. 2020. Generating sports news from live commentary: A Chinese dataset for sports game summarization. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 609–615, Suzhou, China. Association for Computational Linguistics.

Luyang Huang, Shuyang Cao, Nikolaus Parulian, Heng Ji, and Lu Wang. 2021. Efficient attentions for long document summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1419–1436, Online. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Osman Doruk Kicikoglu, Richard Bartle, Jon Chamberlain, Silviu Paun, and Massimo Poesio. 2020. Aggregation driven progression system for GWAPs. In *Workshop on Games and Natural Language Processing*, pages 79–84, Marseille, France. European Language Resources Association.

Kalpesh Krishna, Erin Bransom, Bailey Kuehl, Mohit Iyyer, Pradeep Dasigi, Arman Cohan, and Kyle Lo. 2023. LongEval: Guidelines for human evaluation of faithfulness in long-form summarization. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1650–1669, Dubrovnik, Croatia. Association for Computational Linguistics.

François Lareau, Mark Dras, and Robert Dale. 2011. Detecting interesting event sequences for sports reporting. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 200–205, Nancy, France. Association for Computational Linguistics.

Sha Li, Chi Han, Pengfei Yu, Carl Edwards, Manling Li, Xingyao Wang, Yi Fung, Charles Yu, Joel Tetreault, Eduard Hovy, and Heng Ji. 2023. Defining a new NLP playground. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11932–11951, Singapore. Association for Computational Linguistics.

Bo Liu, Yuqian Jiang, Xiaohan Zhang, Qiang Liu, Shiqi Zhang, Joydeep Biswas, and Peter Stone. 2023a. Llm+p: Empowering large language models with optimal planning proficiency.

Linqing Liu, Patrick Lewis, Sebastian Riedel, and Pontus Stenetorp. 2022. Challenges in generalization in open domain question answering. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2014–2029, Seattle, United States. Association for Computational Linguistics.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023b. Lost in the middle: How language models use long contexts.

Yixin Liu, Alexander R. Fabbri, Jiawen Chen, Yilun Zhao, Simeng Han, Shafiq Joty, Pengfei Liu, Dragomir Radev, Chien-Sheng Wu, and Arman Cohan. 2023c. Benchmarking generation and evaluation capabilities of large language models for instruction controllable summarization.

Yixin Liu, Avi Singh, C. Daniel Freeman, John D. Co-Reyes, and Peter J. Liu. 2023d. Improving large language model fine-tuning for solving math problems.

Angus Loten. 2023. Wendy's, Google train next-generation order taker: an AI Chatbot.

https://www.wsj.com/articles/wendys-google-train-next-generation-order-taker-an-ai-chatbot-968ff865. Accessed on: May 9, 2023.

Stephanie M. Lukin, editor. 2020. *Workshop on Games and Natural Language Processing*. European Language Resources Association, Marseille, France.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. 2023. Mass-editing memory in a transformer.

Jack Merullo, Luke Yeh, Abram Handler, Alvin Grissom II, Brendan O'Connor, and Mohit Iyyer. 2019. Investigating sports commentator bias within a large corpus of American football broadcasts. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6355–6361, Hong Kong, China. Association for Computational Linguistics.

MistralAI. 2023. Mixtral of experts. https://mistral.ai/news/mixtral-of-experts/. Accessed on: December 11, 2023.

OpenAI. 2023. New models and developer products announced at DevDay. https://openai.com/blog/new-models-and-developer-products-announced-at-devday. Accessed on: November 6, 2023.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems?

Ratish Puduppully, Li Dong, and Mirella Lapata. 2019. Data-to-text generation with entity modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2023–2035, Florence, Italy. Association for Computational Linguistics.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning.

Zayne Sprague, Kaj Bostrom, Swarat Chaudhuri, and Greg Durrett. 2022. Natural language deduction with incomplete information. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8230–8258, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Theodore R. Sumers, Shunyu Yao, Karthik Narasimhan, and Thomas L. Griffiths. 2023. Cognitive architectures for language agents.

Chris van der Lee, Emiel Krahmer, and Sander Wubben. 2017. PASS: A Dutch data-to-text system for soccer, targeted towards specific audiences. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 95–104, Santiago de Compostela, Spain. Association for Computational Linguistics.

Dave Van Veen, Cara Van Uden, Louis Blankemeier, Jean-Benoit Delbrouck, Asad Aali, Christian Bluethgen, Anuj Pareek, Malgorzata Polacin, Eduardo Pontes Reis, Anna Seehofnerova, Nidhi Rohatgi, Poonam Hosamani, William Collins, Neera Ahuja, Curtis P. Langlotz, Jason Hom, Sergios Gatidis, John Pauly, and Akshay S. Chaudhari. 2023. Clinical text summarization: Adapting large language models can outperform human experts.

Alex Wang, Richard Yuanzhe Pang, Angelica Chen, Jason Phang, and Samuel R. Bowman. 2022. SQuALITY: Building a long-document summarization dataset the hard way. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1139–1156, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. Association for Computational Linguistics.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. 2023. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts.

Ruochen Xu, Song Wang, Yang Liu, Shuohang Wang, Yichong Xu, Dan Iter, Pengcheng He, Chenguang Zhu, and Michael Zeng. 2023. LMGQS: A large-scale dataset for query-focused summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14764–14776, Singapore. Association for Computational Linguistics.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models.

Jianmin Zhang, Jin-ge Yao, and Xiaojun Wan. 2016. Towards constructing sports news from live text commentary. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1361–1371, Berlin, Germany. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena.