# The Collaborative Abilities of ChatGPT Agents in a Number Guessing Game

Ekaterina Sangati[1†] and Federico Sangati[2] and Marc Slors[3] and Kenji Doya[1]

[1]Neural Computation Unit, Okinawa Institute of Science and Technology, Okinawa, Japan
(kat.sangati@gmail.com)
[2]Cognitive Neurorobotics Research Unit, Okinawa Institute of Science and Technology, Okinawa, Japan
[2]Philosophy of Mind and Language, Radboud University Nijmegen, Nijmegen, The Netherlands

**Abstract:** We present a new behavioral evaluation of ChatGPT-3.5 and 4 Large Language Models. Groups of 3 ChatGPT agents were asked to play a collaborative number-guessing game in which each agent had to submit a number, and the sum of the team's numbers needed to match the target number. ChatGPT-4 model showed better performance than ChatGPT-3.5. However, both models did worse than previously reported human participant results in this game. A deeper analysis of model errors shows that the two models failed for different reasons, and neither model adopted human-like strategies of social coordination.

**Keywords:** Large Language Models; ChatGPT; Collective behavior; Social reasoning; Language model evaluation

## 1. INTRODUCTION

November 2023 marked one year since the public release of ChatGPT – a highly successful artificial intelligence tool that moved so-called Large Language Models (LLMs) from research labs into the public consciousness. Interestingly, within this year, we witnessed claims about not only the impressive *linguistic* abilities of these models but also their emergent general *cognitive* skills such as analogical reasoning [1] or arithmetic [2]. The latter claims are based on new ways of assessing LLM performance that have appeared in the field in tandem with these breakthrough experimental results. Traditionally, language models have been evaluated on standardized linguistic tasks, such as question answering, summarization, part of speech tagging, etc., depending on the goals set out for the model. Now, assessment comes from what some have aptly termed "machine psychology" approach [3], i.e., administering psychological tasks to LLM agents, such as modified versions of Raven's Intelligence test [2], social cognition False Belief test [4] and even tasks in color categorization [5]. Successful results of such experiments can be taken to mean that linguistic input data is sufficient for the development of cognitive abilities in question, which would have significant implications for not just new AI tools but also human cognition. Whether such conclusions are warranted is debated [6, 7]. However, testing LLMs on various tasks beyond their linguistic training is undoubtedly useful as it can give us a broader understanding of what behaviors can be expected from them in real-life use contexts. This is especially important given the increasing participation of LLMs in human society, ranging from help with preparing job applications [8] and research articles [9] to obtaining spiritual advice [10].

In this paper, we propose collaborative games as a methodology to evaluate LLM capacities that offer three advancements compared to previous LLM psychology experiments. First, games present a natural way of testing situated language understanding [11] that goes beyond atomic linguistic or cognitive capacities. Specifically, games provide a controllable form of measuring success on the task, strategies employed, reasoning traces, and collaborative abilities – all within a practical and complete behavioral context.

Second, most of the tasks employed so far have been primarily one-off scenarios. For instance, in a large holistic LLM evaluation project [12], the most common scenario tested is question answering. In such tasks, the model is presented with a question like a natural Google search query and evaluated on the answer quality. However, assessing the performance of *conversational* models requires testing if they can behave adequately over longer-span interaction, i.e., answering appropriately and keeping in line with previous dialog turns (both their own and those of conversation partner). Games typically require such consistency to succeed, constituting a good test bed (cf. [13]).

Finally, the tasks employed so far have been primarily solitary. They involved testing a single LLM agent on language, reasoning, or other cognitive abilities. By contrast, many human capacities are social in nature and are enabled or shaped by interaction. Testing these is especially important when considering scenarios where humans might be asked to collaborate with LLMs in settings beyond a simple human-LLM dyad. Accordingly, this work explores how several LLM agents perform in a collaborative game inspired by a human experiment.

Roberts and Goldstone [14] investigated collective coordination in a number guessing game. In a typical single-player version, a random number within a given range must be guessed in a set number of tries, with feedback on whether the guess is higher or lower than the target. In the collaborative version proposed by Roberts and Goldstone, groups of players submit their guesses and it is their sum that has to match the target. The authors evaluated teams of between 2 and 17 members. The target was a number between 50 and 100, and each player could submit a guess between 0 and 50. Two types of feedback were provided in each game: basic directional feedback (the sum is too low / too high) or more informative numeric feedback, in which the system lets the players know how much the submitted sum differs from the target number. The results showed that people can optimize

† Ekaterina Sangati is the presenter of this paper.

their collective performance, but it is easier accomplished with numeric feedback and in smaller-sized groups. Detailed analysis of behavioral performance also suggested that success partially relies on developing social roles within the group, which enhance mutual predictability between players.

In the current study, we let groups of commercially available LLMs, namely ChatGPT agents [15], play the same social number-guessing game. We study 1) whether they are able to find the solution, 2) how they perform in this task, and 3) whether their performance is similar to strategies adopted by human players. We also compare two specific ChatGPT implementations currently in widespread use: GPT-3.5 and GPT-4. It has been found in many cognitive and behavioral tasks that GPT-4 shows better performance, and hence, we test whether this improvement extends to the collaborative context explored here.

## 2. METHODS

### 2.1. Task and procedure

Given the exploratory nature of this study, we decided to reduce task complexity compared to the game design reported in the human participant study. First, we lowered the number ranges: our targets ranged between 20 and 40, and each individual guess between 0 and 20. Second, we provided the most informative feedback possible: whether the sum is higher or lower than the target number and what number each agent submitted in a given guess attempt. Third, we only tested teams of 3 agents each.

Each team played the game for 6 rounds, and each round had a maximum of 20 attempts to guess the target number. The rules were explained to the agents at the beginning of each game. They were also explicitly told that they were playing with other ChatGPT agents and that this was a collaborative game (see more about this in Section 2.2). Each agent within a team was initialized as a separate ChatGPT instance and had access only to its conversation with the game system. That is, there was no communication possible between ChatGPT agents, and the only information they had about other agents' behavior was what was communicated to them by the system's in-game feedback. Within each round, the whole conversation was provided to the agents to maintain dialog continuity. This message history was not carried over to new rounds to reduce computational and monetary costs. Instead, each round began with the system providing the agent with a tabular summary of the previous rounds. Example conversations from one team per each ChatGPT model are available on GitHub[1].

We tested 10 groups of agents for each model. A list of 10 sequences of 6 target numbers for each team was generated before the game, and we provided the same numbers to both ChatGPT models to avoid any dependence of results on the specific target sequences. The specific models used in this study were gpt-3.5-turbo-0613 and gpt-4-0613 and we refer to them as GPT-3.5 and GPT-4 below.

[1] https://github.com/oist/ChatGptGuessNumber

### 2.2. Prompt design

A significant amount of piloting was required to fine-tune the prompts used in the game. We believe issues encountered in this process are informative for understanding the models' capacities, so we report on them here.

Our initial prompt was limited to explaining that the agents are playing a number guessing game with other agents and that the sum of their guesses must match the target number. We also asked the agents to end their response with a specific pattern that would make it easy to extract the submitted number ("My guess is:"). This resulted in each agent trying to decide for the team rather than guess independently (e.g., "My partner and I choose to guess 10. So, our combined guess is 10."). To remedy this problem, we added qualification in the following prompt iteration that each agent can only offer their individual guess, not a combined sum. The new addition, in turn, prompted the agents to behave individualistically and produce strategies similar to what they would do in an individual version of the game (e.g., starting with the mid-range number: "For the first round, let's start with a conservative guess. Since the range is between 21 and 40, I'll guess 10."). To again facilitate the potential use of social reasoning in the game, the final version of the prompt emphasized that this is a social game and the successful solution relies on joint performance. It also clarified that other team members are ChatGPT agents similarly trying to reason about the best joint strategy. The change resulted in more instances of social reasoning about other agents' moves as described in Section 3.4.

In addition to the initial prompt, we devised in-game messages to handle common errors. One issue frequently encountered with both models was not adhering to the requested response pattern. While some responses could be addressed by extending our range of acceptable expressions (e.g., "My guess for this round will be:"), others contained the numeric answer in the middle of the message and would be difficult to handle. For this reason, whenever the agents did not end their answer with a required pattern, they were asked to restate it. The same strategy was applied to two other errors: offering a number outside the allowed range and responding with a non-integer number. The first error was committed repeatedly by GPT-3.5 agents. We addressed this by modifying our in-game prompt to restate the allowed range ("What's your next guess between 0 and 20?"). The second error, on the other hand, was a common occurrence for GPT-4 agents and was corrected by telling them that floating numbers were not allowed.

### 2.3. Analysis

To quantify performance that takes into account both the number of solved rounds and the number of attempts that lead to the solution, we devised a performance score $P$ calculated with the following formula:

$$P = w_r \cdot \frac{|CR|}{|R|} + w_a \frac{\sum_{r \in CR} 1 - \frac{\alpha(r)-1}{A}}{|CR|}, \tag{1}$$

where $R$ is the set of all rounds, $CR$ is the set of the correct rounds, $A = 20$ is the maximum number of attempts, $\alpha(r)$

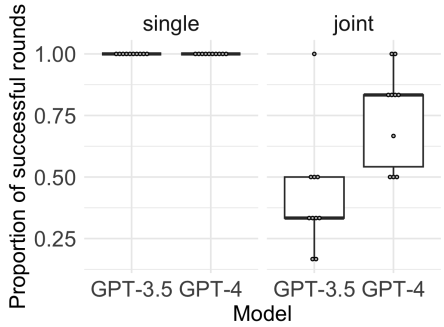


**Fig. 1**. Rounds solved by ChatGPT in individual and joint games.

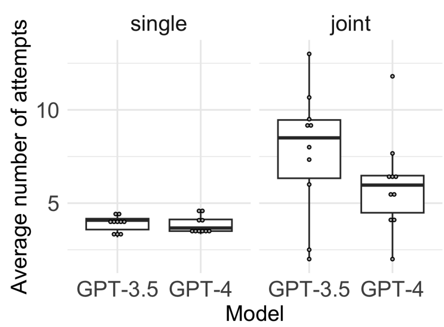


**Fig. 2**. Attempts in successful rounds in individual and joint games.

the number of attempts used to complete the round $r$. The two weights $w_r = 0.6$ and $w_a = 0.4$ determine the importance of succeeding in a round and completing it in as few attempts as possible, respectively. Note that the maximum possible score according to this formula is 1 if the agents solve all the rounds and guess the target number after 1 attempt in each round and 0 if they do not solve any round.

We also followed the original study [14] in implementing a way of quantifying adaptive strategies employed by the agents. Reaction $Rn$ is calculated as $G_a - G_{a-1}$ where $G_a$ is the guess on a given attempt $a$. Reactivity $Rc$ is defined as $G_a - G_{a-1}$ if the group's sum was lower than the target number on a previous attempt and as $G_{a-1} - G_a$ if the sum was higher. This measure is $Rn$ normalized such that positive values are reactions in the direction consistent with the feedback. Moreover, both measures can be computed for the group level sum as the guess $G$ (we will refer to them as $Rn_g$ and $Rc_g$) and for the individual level, i.e., each agent's guess ($Rn_a$, $Rc_a$).

## 3. RESULTS

### 3.1. Individual game

To ensure that the results we obtained in the joint version of the task are not due to fundamental issues with mathematical reasoning or misunderstanding of the rules of the number guessing game, we first run a single-player test with both models ($N = 10$ for each model). The agents were asked to guess a number between 20 and 40 individually and were told in each round if their guess was too low or too high compared to the target number within the same range. Both models were able to find the solution in all 6 rounds (Fig. 1) and did so equally quickly (Fig. 2).

### 3.2. Joint game performance

In the joint version of the game, both models obtained lower scores than in the individual version. Furthermore, groups of GPT-4 agents obtained higher performance scores $P$ ($M = 0.75$) than GPT-3.5 agents ($M = 0.52$), according to a two-sample t-test $t(16.00) = -4.22, p < .001; d = -2.11$, which was due to both more succesful rounds and fewer steps to solution (Fig. 1 and 2). Importantly, ChatGPT performance differed from human results. First, Roberts and Goldstone [14] reported no game round failures and while participants in small groups required similar number of attempts to reach the target ($M = 6.78$, $SD = 1.08$) as our GPT-4 agents, they did so in a more difficult version of the game. Second, neither of the models showed improvement over rounds, contrary to what has been reported for humans.

Figure 3 shows representative examples of game behavior for GPT-3.5 and GPT-4 for the same sequence of target numbers. The models differ in performance and behavioral pat-

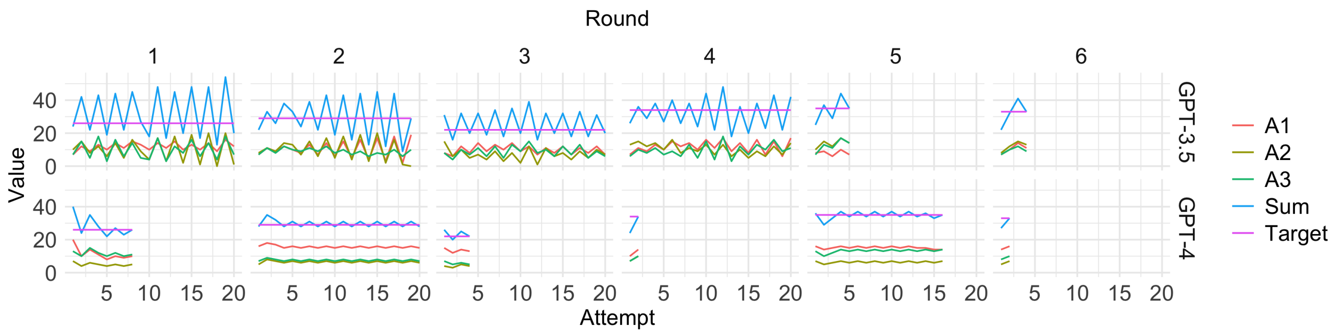


**Fig. 3**. Example of guessing behavior in all game rounds in two groups of agents. The target numbers in these rounds were $26, 29, 22, 34, 35, 33$.

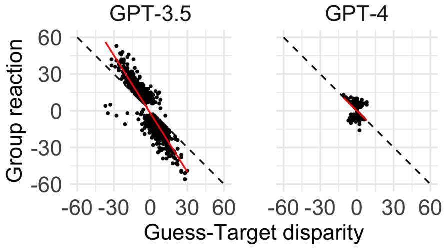


**Fig. 4**. Group reactions compared to the disparity in a given attempt. Dashed lines: the optimal reaction if the disparity is known; red lines: best fit to the model reactions.


**Fig. 5**. Group reactions over subsequent guess attempts. Different colors represent different groups.

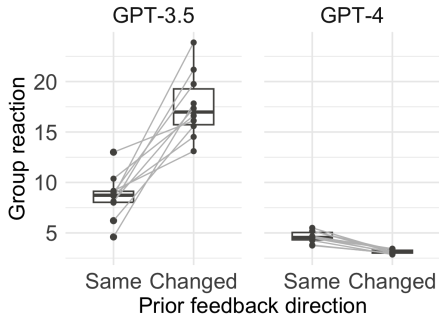


**Fig. 6**. Group reactions in response to change in game feedback.

terns leading to success and failure. While GPT-3.5 seems to fail at finding the solution due to large oscillations around the target number, GPT-4 fails by being trapped in small oscillations and with all group members repeating the same guesses.

### 3.3. Strategy analysis

To better understand the strategies followed by the agents in this game, we can explore how they react from one attempt to the next in response to the system's feedback.

Figure 4 shows group reactions $Rn_g$ plotted against the actual disparity between the guessed sum and the target. It appears that GPT-3.5 agents consistently overreact, as suspected from the example behavior plots. However, the measure for GPT-4 agents shows that their reaction is, in fact, optimal. For comparison, humans are known to underreact in this task (Figure 3 in [14]). Thus, although GPT-4 groups seem to respond in an overall adaptive way, possibly even more adaptive than humans, they still fail on some rounds.

We can now examine how the group reaction changes over subsequent attempts in the game as the agents approach the solution (or the final allowed attempt). If the group can follow an adaptive strategy, it should be 1) decreasing its reaction amount in subsequent attempts and 2) altering the amount of reaction in response to changes in feedback.

Figure 5 shows changes in absolute reaction for two models plotted against steps to the solution, separately for rounds that succeeded and those that failed. This way of aligning responses (instead of just showing subsequent attempts) is more informative, given that the number of attempts differs substantially among groups and game rounds. Just as for the two examples in Figure 3, it can be observed that the source of error for the two models differs for all groups: increasing overreaction for GPT-3.5 and getting stuck in a minimal reaction for GPT-4. By contrast, on successful rounds, GPT-3.5 displays no clear tendency in absolute $R_n$ over steps to solution (repeated measures correlation of $r = -0.02$, *ns.*, 95% CI $[-0.17, 0.12]$). At the same time, GPT-4 adaptively slightly decreases group reaction as the guesses get closer to the target ($r = 0.17$, $p < 0.01$, 95% CI $[0.05, 0.3]$).

Concerning response to feedback, when the system's reply is consistently 'too low' or 'too high', it is adaptive to keep reacting big to approach the target. However, once the feedback direction changes, it is better to decrease the reaction amount. It appears that GPT-4 can follow this strategy while GPT-3.5 increases the reaction in these conditions.

Finally, in addition to group behavior patterns, we can examine whether 1) individual agents become more consistent over time and 2) groups become more diverse over time, which would indicate an emergence of role complementarity. The first tendency can be captured by average within-agent reactivity variance $1/n \cdot \sum_{i=1}^{n} \sigma^2[Rc_{a_i}]$. The second tendency: by between-agent reactivity variance, that is, the variance of average agent reactivities within each group, $\sigma^2[\mu(Rc_{a_1}), \ldots, \mu(Rc_{a_n})]$. These measures can be computed by averaging over rounds or over steps to the solution, where the former would tell us about strategy development as the game progresses, and the latter would be about individual agents' behavior as they approach the solution.

In the human study [14], both adaptive strategies (individual consistency and within-group specialization) held for large groups and were correlated with better performance. In small groups, participants became less consistent and groups became less diverse with time but neither tendency had a relationship with performance level, suggesting that small groups can coordinate well even without specific strategies.

For GPT-3.5, the within-agent variance does not change

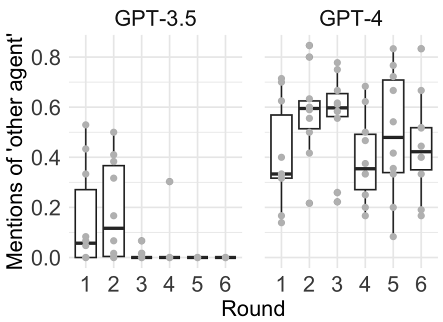


**Fig. 7**. Group reactions.

over rounds, while between-agent variances decreases significantly ($r = -0.3, p < 0.05$). For GPT-4, within-agent variance reduces significantly over rounds ($r = -0.31, p < 0.05$), i.e., each agent becomes more consistent, but there is no change in between-agent variance. Furthermore, both within and between-agent variances are significantly lower overall for GPT-4 groups. These patterns could be cautiously interpreted as GPT-3.5 groups becoming more homogeneous while GPT-4 remaining homogeneous throughout the game.

When averaged over steps to the solution instead of over rounds, the patterns of reactivity variances are consistent with the nature of maladaptive strategies for each model. Specifically, for GPT-3.5, there is a significant increase in within- ($r = 0.87, p < 0.001$) and between-agent ($r = 0.58, p < 0.001$) reactivity in the failed rounds, which parallels our observation that this model suffers from group over-reaction as the attempts progress. For GPT-4, individual agents become less consistent over attempts in successful rounds ($r = 0.34, p < 0.05$) but more consistent in the failed rounds ($r = -0.3, p < 0.01$). In addition, their groups also become more homogeneous in the failed rounds ($r = -0.37, p < 0.001$), which mirrors the general GPT-4 problem with a lack of individual diversity.

### 3.4. Social reasoning

Our final analysis considers the evidence for social reasoning in ChatGPT agents. Our prompt emphasized the need for a joint strategy but did not explicitly request the agents to explain their reasoning. Nevertheless, both models offered explanations for their guesses, at least initially. GPT-3.5 agents did so in the first 2-3 rounds but then uniformly switched to only stating their guess, while GPT-4 agents continued justifying their guess choices. Interestingly, as part of these explanations, the answers of both models contained statements that indicated some concern about other agents' guesses and strategies. As a proxy measure for this behavior, we counted the number of times the expression "other agent" appeared in the message string. Figure 7 shows each pair's average number of social mentions per attempt over the game rounds. The agents did not modify the frequency of their social mentions as the game progressed. There was also no relationship between the average number of social mentions in each attempt

**Table 1**. Examples of ChatGPT guessing messages in different pairs (PairID), rounds (R), and guess attempts (A).

| Model | PairID | R | A | Answer |
|---|---|---|---|---|
| GPT-3.5 | 46698 | 1 | 1 | In order to maximize our chances of guessing the correct number, I suggest we start with a conservative strategy. Let's try to evenly distribute our guesses between 0 and 20. |
| GPT-3.5 | 46698 | 1 | 6 | Based on the fact that the previous sum was too low and the guesses of the other agents were 7 and 6, I will adjust my guess accordingly for the next round. |
| GPT-3.5 | 86411 | 2 | 1 | My strategy in this round will be to provide a diverse range of guesses while considering the previous guesses made by the other agents. This way, we can explore different possibilities and increase our chances of finding the correct sum. |
| GPT-3.5 | 229107 | 2 | 3 | Based on the previous feedback, it seems like the other agents have increased their guesses. To balance this out, I will decrease my next guess. |
| GPT-4 | 901047 | 1 | 1 | If I guess around 7, it gives us some leeway if my estimate is off and also allows the other two agents to adjust their guesses based on mine. |
| GPT-4 | 86411 | 4 | 2 | I anticipate that the other ChatGPT agents might also increase their guesses in response to the last feedback. Therefore, I'll choose a modest increase. |
| GPT-4 | 901047 | 3 | 1 | Based on the earlier rounds, it seems that the other two agents are guessing relatively high. I will try to counterbalance this by guessing lower. |
| GPT-4 | 46698 | 1 | 17 | I understand that we are stuck in a cycle, and the other agents are apparently maintaining their guesses. It's clear that the target number is between 19 and 22. Since my previous guess of 12 resulted in the total being too low, I will increase my guess slightly to try and adjust the total sum nearer the target. |

and behavioral success. Specifically, this measure of sociality did not differ between correctly solved and failed rounds and did not correlate with solution speed in correct rounds.

This quantitative result is perhaps not surprising given that a mere mention of other agents does not guarantee the quality of social reasoning. Table 1 lists several specific examples of guess explanations offered by agents. GPT-3.5 does try to consider how other agents behave (Pairs 86411 and 229107 examples). Still, most of its references are what we would consider 1st order social mentions - considering other agents as constraints in the game. When GPT-4 mentions other agents, it also reflects on how those agents might behave in response to their own behavior or game feedback, which would be more akin to 2nd order social reasoning. It also reasons about its partners' long-term game behavioral patterns. Unfortunately, even when GPT-4 recognizes a problem, like being stuck in a guessing loop (Pair 46698), it is unable to correct it. Specifically, the example in Table 1 shows a response from Agent $A_1$ in Attempt 17 that comes in the middle of a "too high - too low" loop with a guess sum oscillating between 19 and 22. It can correctly identify these boundaries of the target number and also correctly decides to change its guess. However, it changes it with respect to the immediately previous response, i.e., $G_{a-1}$ without realizing that the loop has a length of 2, and therefore it should change the guess with respect to $G_{a-2}$. With the other two agents committing the same error, the group fails to exit the loop.

Our study aimed to shed light on ChatGPT performance when placed in a complete behavioral context with little external intervention. However, it is possible that with additional prompting from the game system when an interactive loop occurs, the agents would be able to generate better solutions.

## 4. DISCUSSION

We have proposed a new collaborative game evaluation of LLM models and applied it to two versions of ChatGPT. The game is based on an existing human experiment, and the differences between human and ChatGPT participants' behavior are instructive.

First, we found that both models are able to solve the individual number guessing game but do not reach human performance levels in the joint setting. Therefore, any deficiency in joint number guessing is not due to mathematical reasoning problems but due to collaborative challenges – needing to coordinate guesses with team members. This coordination requires both reasoning about other agents' behavior (recognizing their previous guess patterns and anticipating future guesses), as well as adjusting one's own behavior to make it easier for other agents to predict it. Humans do this by a set of social cognitive capacities and behavioral strategies such as adopting consistent and complementary roles within a group. Our results show that ChatGPT agents are not acting in a similar manner.

Second, lack of improvement over subsequent rounds and problematic entrapment in oscillations show that ChatGPT is not able to behave adaptively in time spans longer than a single conversational turn. No change over the course of the game, such as the development of division of labor strategy, is perhaps not surprising given that these models are not equipped with learning capacity. The inflexibility over several turns, however, is concerning because it could be expected on the basis of conversational coherence. At the same time, this result is consistent with preliminary findings by [13] that ChatGPT has difficulty tracking belief states in multi-turn dialogs.

Finally, we observed a lack of diversity in behavior among the agents, especially in GPT-4 model groups. It is likely that this has contributed to overall inflexibility in strategies that led to failure on some rounds of the game. In future research, we plan to test whether creating teams out of agents with different parameters, or endowing team members with different "personalities" might help them overcome this problem.

Overall, GPT-4 showed reasonable performance on a challenging collaborative task. The fact that it does not show human-like behavior should be kept in mind when designing interactions with human users. On the one hand, mechanical lack of variability and consistency might be a desirable feature of such models as it would make them more predictable. On the other hand, some aspects of interaction witnessed in our results, like stubborn repetition of the same answers, might be frustrating. The present results do not tell us how ChatGPT would perform in such a mixed human-machine scenario nor how it would be perceived, and that is an exciting avenue for future studies.

## REFERENCES

[1] T. Webb, K. J. Holyoak, H. Lu. "Emergent analogical reasoning in large language models." Nature Human Behaviour, Vol. 7, No. 9, pp. 1526–1541, 2023.

[2] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, others. "Emergent abilities of large language models." arXiv:2206.07682 [cs.CL], 2022.

[3] T. Hagendorff. "Machine psychology: Investigating emergent capabilities and behavior in large language models using psychological methods." arXiv:2303.13988 [cs.CL], 2023.

[4] S. Trott, C. Jones, T. Chang, J. Michaelov, B. Bergen. "Do Large Language Models Know What Humans Know?" Cognitive Science, Vol. 47, No. 7, p. e13309, 2023.

[5] R. Marjieh, I. Sucholutsky, P. van Rijn, N. Jacoby, T. L. Griffiths. "Large language models predict human sensory judgments across six modalities." arXiv:2302.01308 [cs.CL], 2023.

[6] I. A. Blank. "What are large language models supposed to model?" Trends in Cognitive Sciences, 2023.

[7] I. van Rooij, O. Guest, F. G. Adolfi, R. de Haan, A. Kolokolova, others. "Reclaiming AI as a theoretical tool for cognitive science." arxiv.org, 2023.

[8] K. Kaneko. "Job applicants in Japan embrace ChatGPT to improve their chances." https://www.japantimes.co.jp/news/2023/06/29/national/chatgpt-job-applicant-use/, 2023.

[9] G. Conroy. "Scientists used ChatGPT to generate an entire paper from scratch — but is it any good?" Nature, Vol. 619, pp. 443–444, 2023.

[10] N. S. Fisher. "Generative AI Has Ushered In the Next Phase of Digital Spirituality." Wired, 2023.

[11] D. Schlangen. "Dialogue Games for Benchmarking Language Understanding: Motivation, Taxonomy, Strategy." arXiv:2304.07007 [cs.CL], 2023.

[12] R. Bommasani, P. Liang, T. Lee. "Holistic Evaluation of Language Models." Annals of the New York Academy of Sciences, Vol. 1525, No. 1, pp. 140–146, 2023.

[13] Y. Bang, S. Cahyawijaya, N. Lee, W. Dai, D. Su, others. "A Multitask, Multilingual, Multimodal Evaluation of ChatGPT on Reasoning, Hallucination, and Interactivity." arXiv:2302.04023 [cs.CL], 2023.

[14] M. E. Roberts, R. L. Goldstone. "Adaptive group coordination and role differentiation." PloS One, Vol. 6, No. 7, p. e22377, 2011.

[15] OpenAI. "GPT-4 technical report." Tech. Rep. CoRR, abs/2303.08774, 2023.