

# A Notion of Complexity for Theory of Mind via Discrete World Models

Anonymous ACL submission

## Abstract

Theory of Mind (ToM) can be used to assess the capabilities of Large Language Models (LLMs) in complex scenarios where social reasoning is required. While the research community has proposed many ToM benchmarks, their hardness varies greatly, and their complexity is not well defined. This work proposes a framework to measure the complexity of ToM tasks. We quantify a problem’s complexity as the number of states necessary to solve it correctly. Our complexity measure also accounts for spurious states of a ToM problem designed to make it apparently harder. We use our method to assess the complexity of five widely adopted ToM benchmarks. On top of this framework, we design a prompting technique that augments the information available to a model with a description of how the environment changes with the agents’ interactions. We name this technique Discrete World Models (DWM) and show how it elicits superior performance on ToM tasks.<sup>1</sup>

## 1 Introduction

Theory of Mind (ToM) studies how agents form and use beliefs to reason in dynamic environments (Premack and Woodruff, 1978). Originally developed to describe human interactions (Preston and De Waal, 2002; Tomasello, 2009) as well as toddlers’ psychological development (Wimmer and Perner, 1983; Baron-Cohen et al., 1985), ToM has been quickly adopted by other fields, including artificial intelligence (McCarthy, 1979; Scasselati, 2002) and machine learning (Rabinowitz et al., 2018). In machine learning, ToM has both descriptive and prescriptive usage: on the one hand, ToM benchmarks assess the capabilities of a model in complex environments; on the other, ToM’s frameworks such as *theory-theory* (Gopnik and Wellman, 1994) and *simulation theory* (Churchland, 2013)

<sup>1</sup>Code and data for **full reproducibility** are available in the Code Material.

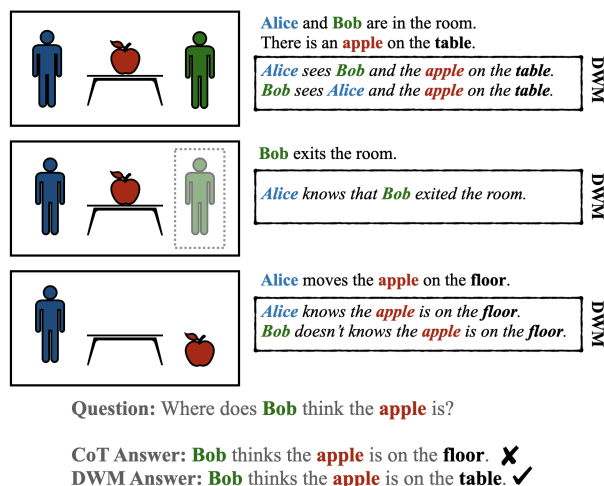


Figure 1: Example of the DWM prompting technique on a classical Sally-Anne QA task (Baron-Cohen et al., 1985). Inspired by our complexity framework (Section 3.1), DWM takes the original task and splits it into sequences, the *state events* (see Def. 3.1), and prompts the LLMs to describe the states. We show that, in most cases, this aids the LLM in providing correct answers.

have been widely adopted to test the proficiency of Large Language Models (LLMs) in social tasks where humans excel (Strachan et al., 2024).

In this work, we propose a framework to characterise a ToM benchmark’s difficulty, i.e., its **complexity**, as the number of *state events* that are sufficient to track the state of an object, including  $k^{\text{th}}$ -order beliefs.

We characterise the complexity of five standard ToM benchmarks, from false belief to common-sense and social reasoning, and compute their complexity as a proxy of their inherent difficulty. Inspired by prompting techniques that split a task into elementary sub-problems that are solved sequentially, like Tree of Thoughts (Yao et al., 2023) and least-to-most prompting (Zhou et al., 2023a), we introduce a technique that stimulates a model’s reasoning capabilities via Discrete World Models (DWM). DWM leverages the notion of state-

fulness via a succinct and coherent representation of each *state events*, as illustrated in Figure 1. We test DWM on ToMi (Le et al., 2019), MindGames (Sileo and Lernould, 2023), Adv-CSFB (Shapira et al., 2023), SocialIQA (Sap et al., 2019), and FANToM (Kim et al., 2023), eliciting superior performance than Chain of Thoughts (CoT) (Wei et al., 2022) and Tree of Thoughts (ToT) (Yao et al., 2023) on those problems whose *state spaces* are informative. We further assess whether memorisation affects a model’s performance, and we discover that while this phenomenon happens for standard benchmarks such as ToMi (Le et al., 2019), with input-output pairs that can be retrieved *word for word* via prompting, it does not correlate with a drop of performance. We conduct our experiments on a variety of open- and closed-source LLMs, including GPT-3.5-Turbo, GPT-4 (OpenAI, 2023), LLaMA3-70B (AI@Meta, 2024) and Mixtral 8x7B (Jiang et al., 2024). In summary, in this paper:

- We introduce the concept of **complexity** of a ToM task to quantify the hardness of keeping track of the elements (e.g., agents’ beliefs or objects’ states) that are sufficient to produce the correct answer to different problems.
- We propose DWM, a simple yet effective prompting technique that improves a model’s capability by making **implicit** information explicit while not necessitating *exogenous information* (i.e., it does not require RAG or fine-tuning).

We consider our work a step towards a framework that formalizes the hardness of a ToM problem univocally inspired by the theory of World Models (Wong et al., 2023).

## 2 Related Work

Over 40 years of research on ToM in psychology (Premack and Woodruff, 1978; Baron-Cohen et al., 1985; Dennett, 1988; Wellman, 2017) on human development has created a fertile ground for the development of these ideas in adjacent fields. In the last decade, many works studied ToM in artificial intelligence and machine learning (Baker et al., 2011; Rabinowitz et al., 2018), with applications to multi-agent systems and reinforcement learning (Gronauer and Diepold, 2022). More recently, the rise in popularity of LLMs shifted the interest towards understanding and benchmarking large

models’ capacity to solve increasingly complex ToM tasks (Aru et al., 2023; Zhou et al., 2023b; Mahowald et al., 2024). While some researchers believe LLMs have already become proficient in solving ToM tasks (Bubeck et al., 2023; Kosinski, 2023; Strachan et al., 2024), others show scepticism and illustrate cases where they fail on trivial variations of well-known problems (Ullman, 2023; Shapira et al., 2023; Sap et al., 2023). In a joint effort between computer scientists and psychologists, many ToM benchmarks have been developed and used to test neural-network models, including LLMs (Gandhi et al., 2022; Chen et al., 2024; Strachan et al., 2024). Recently, concepts such as World Models (Ha and Schmidhuber, 2018) have found applicability and mostly as discrete prompting techniques in conjunction with optimisation procedures (Hao et al., 2023; Moghaddam and Honey, 2023). Researchers have found evidence of an emergent internal representation (e.g., World Model’s surrogates) of the state games (Li et al., 2022; Toshniwal et al., 2021) and state-tracking abilities (Li et al., 2021; Kim and Schuster, 2023; Kim et al., 2024), necessary for correct belief tracking in ToM problems. The works more similar to our complexity framework are only tangentially related to ToM. Inspired by the work in (Zhou et al., 2023a) and the results in (Zhou et al., 2023b). Our prompting technique is inspired by (Park et al., 2023) and (Nye et al., 2021): the former develops an architecture to record the agent’s experiences. The latter proposes a prompting technique that forces a model to express the intermediate computational steps to solve a problem.

## 3 Methodology

In this section, we introduce a notion of complexity for ToM problems: such notion quantifies the hardness of a problem as the number of *computational steps* humans take to solve them. We then present the DWM prompting technique within the complexity framework and show how it differs from standard methods like CoT and ToT. We further characterise its efficiency with the number of input/output tokens and queries to a model as the control variables.

### 3.1 On the Complexity of ToM

Providing a consistent representation of the environment, including each agent’s beliefs, inspired us to characterise the complexity of a ToM problem

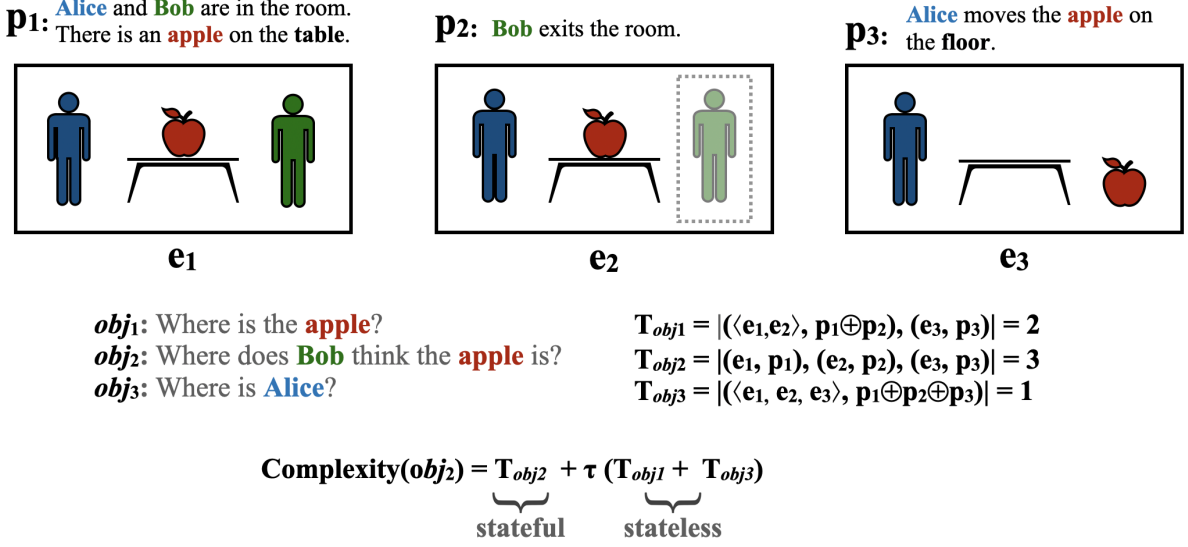


Figure 2: How statefulness and statelessness (Def. 3.2) are computed for the motivating example in Fig. 1. For *obj*<sub>1</sub>, an optimal split to track the **apple** merges the first two states and chunks of the input prompt. For *obj*<sub>2</sub>, which involves the 1<sup>st</sup>-order belief of **Bob**, the statefulness is higher, with *e*<sub>2</sub> that cannot be merged with *e*<sub>3</sub> as it introduces partial observability. The complexity of the task (bottom) is computed as per Eq. 2, with the complexity of stateless objects that is discounted as not directly relevant to the question/answer.

in terms of **sufficient elements to track** to output the correct result. Consider a problem prompt *p*, expressed in natural language, that describes how multiple agents interact with an environment object *obj*, as illustrated in Figure 2 (top). In our framework, an object can be the state of the apple as well as the *k*<sup>th</sup>-order belief of an agent about the apple position. Our framework naturally extends to multiple objects by considering their union.

Suppose that in *p*, the state of *obj* is modified  $T > 0$  times, thus identifying  $T$  unique configurations, namely  $E_{obj} = \{e_1, \dots, e_T\}$ . To correctly solve a ToM task where *p* is complemented by a query about *obj*, a model should distinguish between the interactions that modify the configuration of *obj*, i.e., the **stateful** states, from those that modify any other **stateless** object  $Obj \setminus obj$ , i.e., those one does not need to track.

We first show how to define the cost of tracking a task’s stateful states, which we complement with that of the stateless. Both definitions concur in defining the **complexity** of a ToM task.

### 3.1.1 Stateful and Stateless Complexity

For a ToM task, expressed as *p*, that describes the evolution of an environment where an unknown number of atomic iterations  $T$  modifies *obj* or its perception, each environment state  $e_t \in E_{obj}$  can be coupled with the prompt prefix  $p_{\leq t}$  s.t.  $p_{\leq t} \oplus$

$p_{>t} = p$ , that describes such configuration. We denote  $(e_t, p_{\leq t})$  as a generic *state description*, as illustrated in Figure 2 (top).

**Definition 3.1** (State event). A *state event* for an object *obj* is an event that links adjacent *state descriptions* that involve, for both the environment state  $e_t$  and the sub-prompt  $p_{\leq t}$ , a state change of *obj*. Formally, we define a relation,  $F_{obj}$ , to specify which pairs of state descriptions form a state event:  $F_{obj}((e_t, p_{\leq t}), (e_{t+1}, p_{\leq t+1})) \equiv e_t \neq e_{t+1} \wedge p_{\leq t+1} = p_{\leq t} \oplus p_{t+1}$  where  $1 \leq t \leq |p|$ . ( $|p|$  denotes the number of atomic prompts.)

Thus a *state event*  $F_{obj}$  identifies those *state descriptions*  $(e_t, p_{\leq t})$  which have a successor  $(e_{t+1}, p_{\leq t+1})$  where *obj* has changed its configuration.

In the context of ToM tasks, a *state event* could be a person who moves an object, exits (thus introducing partial observability) or witnesses a change in the environment (as now the description of the environment will take that change into account), as illustrated Figure 2 (middle). Our prompting technique, namely DWM (Section 3.2.1), aims at making implicit observations about objects explicit.

We finally introduce the notion of *partition function* to connect the **maximum number** of non-empty *state events* relative to a prompt. Such a notion will serve as the building block to compute the complexity of a ToM problem.

**Definition 3.2** (Partitions). A *partition*  $part_{\mathbf{obj}}$  w.r.t.  $\mathbf{obj}$  identifies those *state events* which partition a ToM prompt  $p$  into sequential segments where  $\mathbf{obj}$  changes its value. Formally:

$$\text{Let } part_{\mathbf{obj}} = \{(e_t, p_{\leq t}) : F_{\mathbf{obj}}((e_t, p_{\leq t}), (e_{t+1}, p_{\leq t+1})) \wedge e_t \in E_{\mathbf{obj}}\} \quad (1)$$

Def. 3.2 describes an optimal partition,  $part_{\mathbf{obj}}$  of *state descriptions* that covers all the relevant changes to  $\mathbf{obj}$ . The partition is represented by the set of event descriptions where  $\mathbf{obj}$  changes its description immediately after. Note that this set of event descriptions is unique for any  $\mathbf{obj}$ .

### 3.1.2 The Complexity of a ToM Task

We can now define the notion of **statefulness** of a ToM task specified as a prompt  $p$  as the size of Eq. 3.2, namely  $T_{\mathbf{obj}} = |E_{\mathbf{obj}}|$ . The process of computing the statefulness of an object or its belief is illustrated in Fig. 2.

For a ToM task where the question to solve relates to an object  $\mathbf{obj}$ , one must ensure that changes to any other object, namely  $Obj \setminus \mathbf{obj}$ , do not affect  $\mathbf{obj}$ . While tracking the evolution of what is irrelevant to answer the question is unnecessary, a computation model must assess whether a particular environmental change affected  $\mathbf{obj}$ . We thus introduce the notion of **statelessness**, i.e., the cost of discerning whether a change in the environment affects  $\mathbf{obj}$ . The computation is similar to that of Def. 3.2, with  $\mathbf{obj}$  that is replaced by any object in  $Obj \setminus \mathbf{obj}$ ; yet, for stateless objects, we introduce a discount factor  $\tau$  to penalise the complexity of *state events* that do not affect  $\mathbf{obj}$ . Mathematically, we formalise the statelessness of a ToM task involving an object  $\mathbf{obj}$  as  $\tau \sum_{obj \in Obj \setminus \mathbf{obj}} T_{obj}$ .

Finally, we formalise the complexity of a ToM task w.r.t. an object  $\mathbf{obj}$  as the complexity of the stateful states plus the (discounted) sum of the others (i.e., stateless). Namely:

$$T_{\mathbf{obj}} + \tau \sum_{obj \in Obj \setminus \mathbf{obj}} T_{obj} \quad (2)$$

The process of computing the complexity of a ToM task is illustrated in Figure 2.

## 3.2 Discrete World Models

We first introduce the background notation for prompting LLMs and assessing their accuracy on a standard classification task. We then propose

our technique, namely DWM, which we eventually connect with the notion of statefulness of a ToM task.

**Background notation.** A (Large) Language Model is a function that predicts the next token (out of a finite vocabulary) conditioned on the sequence of previously fed/generated tokens, namely  $\psi : \mathbf{v} \in V^* \rightarrow v \in V$ . Such a mechanism can be used to sample multiple token outputs until an ‘end-of-text’ token is predicted by invoking  $\psi$  in an auto-regressive fashion, i.e.,  $\psi(v|\mathbf{v})$ . In our setting, a problem is specified as a tuple  $(p, Q)$ , where  $p$  is a ToM problem and  $Q$  is a *query* function that modifies  $p$  according to a prompting technique, namely  $Q : p \rightarrow p'$ . The LLMs output  $y$  for an input  $Q(p)$  is then compared for correctness against an oracle  $\Omega$ , i.e.,  $\Omega : \psi(Q(p)) \rightarrow \{0, 1\}$ , where 1 means correct classification (0, otherwise). On a sample of  $N > 0$  ToM problems, the accuracy of a model  $\psi$  is then measured as  $\frac{1}{N} \sum_{i=1}^N \Omega(\psi(Q(p_i)))$ , i.e., the average number of times a model is correct in its prediction.

### 3.2.1 Discrete World Models via Prompting

Given a ToM problem  $p$  and a constant  $T \leq |p|$ , we can rewrite  $p$  as  $p_1 \oplus p_2 \oplus \dots \oplus p_T$ . Our *query* function adds a standard preamble  $x$  similar to that of CoT. DWM inserts, after each “split”  $p_t$ , an additional prompt  $w$  like ‘Now, provide a succinct description of the state of the environment and each agent’s belief.’ and query an LLM to provide a representation of the current *state description* of the environment. An LLM is initially queried with  $x \oplus p_1 \oplus w$ , and the answer  $a_1$  is concatenated to the next query, i.e.,  $\psi(x \oplus p_1 \oplus w \oplus a_1 \oplus p_2 \oplus w)$  to retrieve  $a_2$ . The process is carried on for each of the  $T$  chunks, and, at the end,  $y$  is concatenated to eventually prompt the model for the correct answer to  $p$ .

Let  $z_1 = \psi(x \oplus p_1)$ ,  $z_t = \psi(x \oplus p_1 \oplus z_1 \oplus p_2 \oplus \dots \oplus z_{t-1} \oplus p_t) = \psi(x \oplus (\bigoplus_{i=1}^{t-1} p_i \oplus z_i) \oplus p_t)$ , then, the final query is

$$\psi(x \oplus \left( \bigoplus_{t=1}^T p_t \oplus z_t \right) \oplus y) \quad (3)$$

In this sense, our **partition function** (Def. 3.2) consists of splitting a prompt into sequential chunks of the prompt, while the LLM is prompted to provide each *state event* at time  $1 \leq t < T$  as  $e_t = \psi(x \oplus (\bigoplus_{t'=1}^t p_{t'} \oplus z_{t'})) \oplus \omega$ . The process

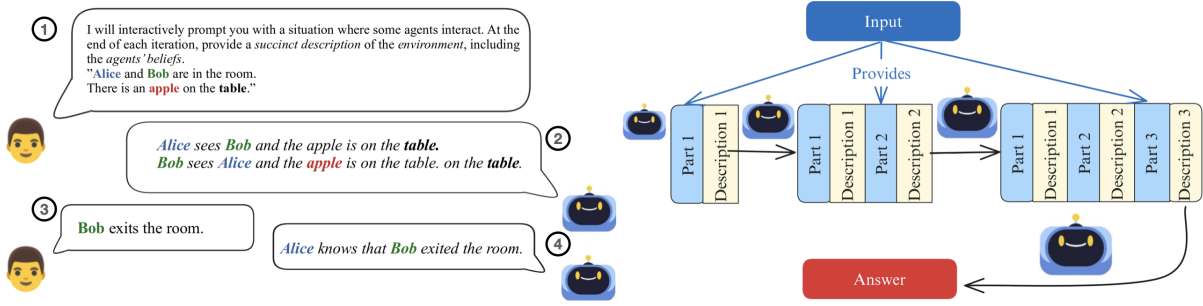


Figure 3: Left: illustration of DWM prompting as per the example in Figure 1. We interactively prompt an LLM with a ToM problem, asking to provide a succinct representation of each agent’s beliefs. Right: schematic presentation of the DWM method. We first break the input string into  $T$  state descriptions. Then, for each part, we ask the LLM to provide the **state event** of the environment and how it changes. In the last step, every part of the input and description is fed to the LLM with another prompt to get the answer for the task.

of prompting a model with DWM is illustrated in Figure 3.

### 3.2.2 On the Complexity of DWM

DWM progressively calls an LLM  $T > 0$  times to generate informative states. For a ToM problem of length  $n$  (i.e., the number of input tokens), which we assume, w.l.o.g., that can be split into  $k$  chunks of approximately the same length  $|x \oplus p_i \oplus w| = \frac{n}{T}$ , the number of tokens generated by an LLM is in the order of  $\mathcal{O}(\sum_{t=1}^T |x \oplus p_t \oplus w \oplus a_t|^2)$ , where  $p_t$  ( $a_t$ ) is the portion of the problem (answer) prompted (retrieved) at iteration  $t$ . With the further assumption that each answer retrieved at split  $t \leq T$  has the same length  $o$ , the complexity is further simplified to be asymptotic to  $\mathcal{O}((\frac{n}{T} + o)^2)$ . Compared to CoT, whose complexity is  $\mathcal{O}(n+o)$ , DWM requires an additional linear number of calls to the model. On the other hand, ToT with the same number of splits  $\frac{n}{T}$  and  $m > 1$  experts results in even higher complexity, i.e., asymptotic to  $\mathcal{O}(m(\frac{n}{T} + o)^2)$ .

## 4 Experimental Evaluation

The experiments are organised as follows. We first test the performance of DWM on ToMi (Le et al., 2019), MindGames (Sileo and Lernould, 2023), Adv-CSFB (Shapira et al., 2023), SocialIQA (Sap et al., 2019), and FANToM (Kim et al., 2023), comparing it with CoT (Wei et al., 2022), ToT (Yao et al., 2023) and prompting with structured data (struct), i.e., the model is queried to first represent the problem in a structured format such as JSON or Yaml. We further show that ToMi has been memorised *word for word* by GPT models, with CoT (and any technique that leaves the input unchanged) being the best-performing method. We

then quantify the complexity of the benchmarks introduced above and highlight the correlation with the models’ performances. Our framework shows complexity ranges between easy and hard problems, even within a benchmark. We conduct our experiments on GPT-3.5-Turbo, GPT-4 (OpenAI, 2023), LLaMA3-70B (AI@Meta, 2024) and Mixtral 8x7B (Jiang et al., 2024).

### 4.1 DWM on ToM Benchmarks

We report results for GPT-3.5-Turbo and Mixtral 8x7B on the five ToM benchmarks: for reasons of space, results for LLaMA3-8B, LLaMA3-70B and GPT-4 are reported in the Appendix, Section B.1. As illustrated in Figure 4 (top), DWM improves the performance of GPT-3.5-Turbo on Mindgames, FANToM and Adv-CSFB by a solid margin. On SocialIQA, which has very short inputs, DWM performs slightly worse than CoT but better than ToT. On the other hand, on ToMi, the best prompting techniques are CoT and ToT. As analysed in the next section, we believe this not be caused by memorisation, which we prove happening, but by the inherent complexity of the task. With Mixtral 8x7B (Fig. 4 (bottom)), DWM improves the performance on ADVcsfb, FANToM, ToMi and Mindgames, and pairs that of CoT on SocialIQA.

**DWM elicits more informed state spaces.** We qualitatively analysed the information elicited by an LLM when prompted with DWM and discovered that it forces a model to output information **not explicitly** available in the prompt. Consider the ToMi example in Figure 5 where GPT-4 is prompted with a situation where agents interact

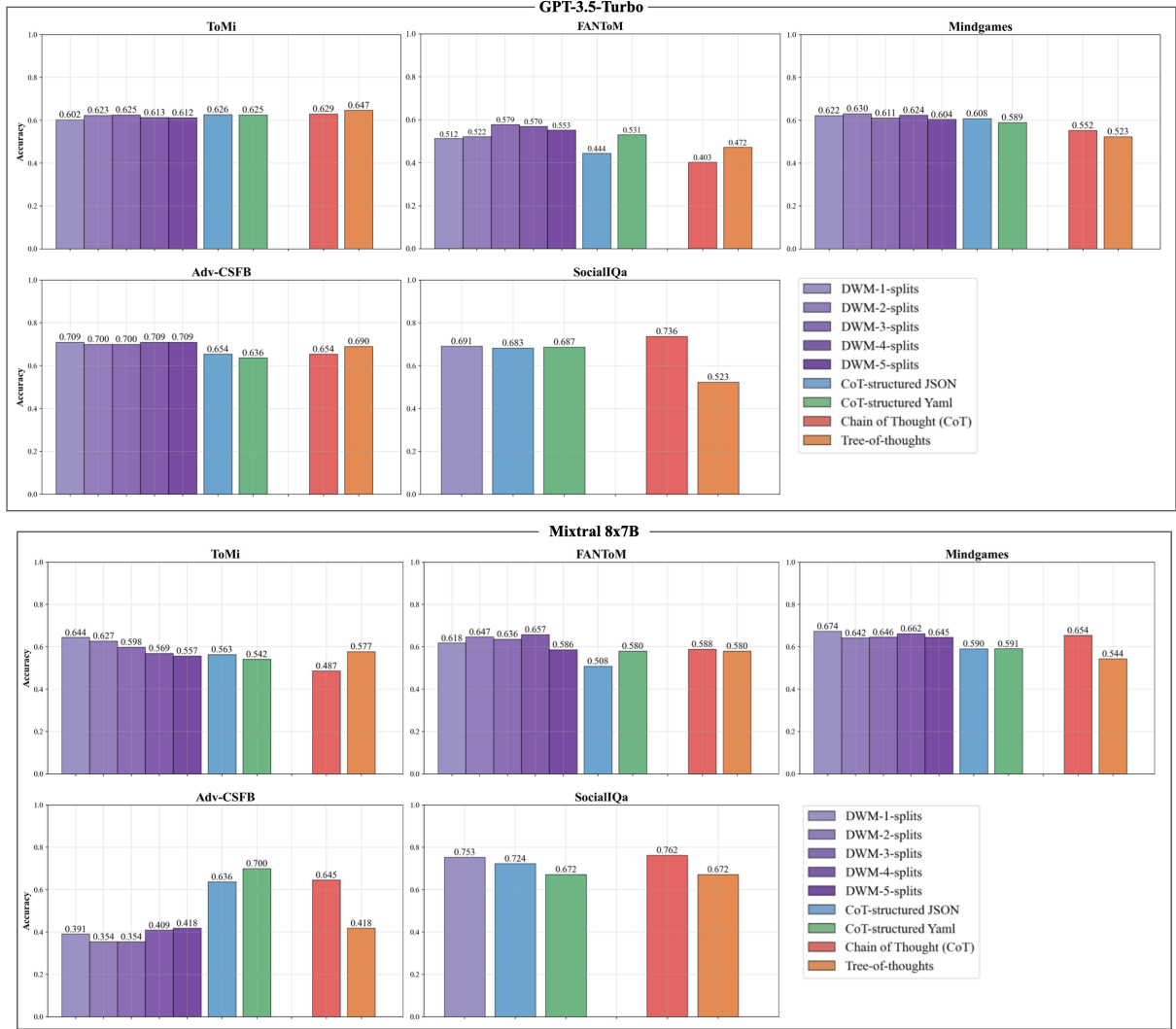


Figure 4: Benchmarks of GPT-3.5-Turbo (top) and Mixtral 8x7B (bottom) models on different ToM tasks for DWM (one to five splits), CoT, ToT and structured prompts (JSON and Yaml).

	ToMi	FANToM	Mindgames	Adv-CSFB	SocialIqa
<b>Memorisation - perfect match</b>	52%	35%	2%	0%	0%
<b>Memorisation - fuzzy</b>	89 ± 15%	74 ± 24%	64 ± 18%	51 ± 11%	40 ± 12%
<b>DWM</b>	0.625	<b>0.579</b>	<b>0.618</b>	<b>0.8364</b>	0.691
<b>CoT</b>	<b>0.629</b>	0.403	0.552	0.7091	<b>0.736</b>

Table 1: Summary of the memorisation test on five ToM benchmarks. We prompted GPT-3.5-Instruct to predict the continuation of 100 randomly sampled test points. We computed the exact and fuzzy memorisation rate (second row, similarity score computed via the Levenshtein distance, see the [thefuzz](#) package), which we complement with the best performance across models of CoT and DWM.

	ToMi	FANToM	Mindgames	Adv-CSFB	SocialIqa
<b>Statefulness</b>	2.62 ± 1.68	2.44 ± 0.96	1.22 ± 0.90	3.24 ± 1.35	1. ± 0.
<b>Statelessness</b>	4.27 ± 2.1	59.42 ± 18.91	5.24 ± 2.71	2.86 ± 1.34	1.14 ± 0.447
<b>DWM - Best Split</b>	3	3	1	4	1

Table 2: Summary of the statefulness and statelessness of different ToM benchmarks. At the bottom, the value of the split that guarantees max performance of GPT-3.5-Turbo with DWM, which we notice is strongly correlated with the statefulness of each benchmark.

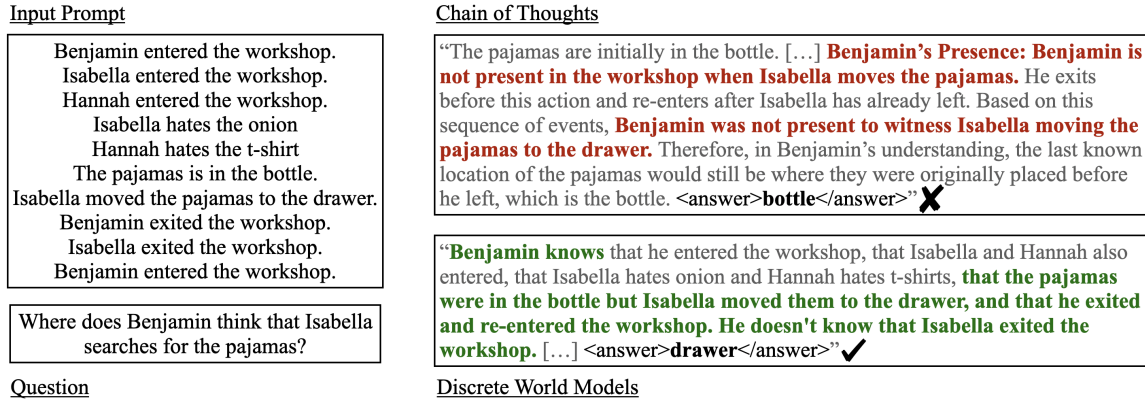


Figure 5: Example of a real ToMi example where GPT-4 fails when prompted with CoT, yet succeeds with DWM. CoT elicits an untruthful reasoning process (in red), while DWM correctly informs the model with the implicit information about Benjamin’s first-order belief (in green). More examples are reported in the Appendix, Section B.2.

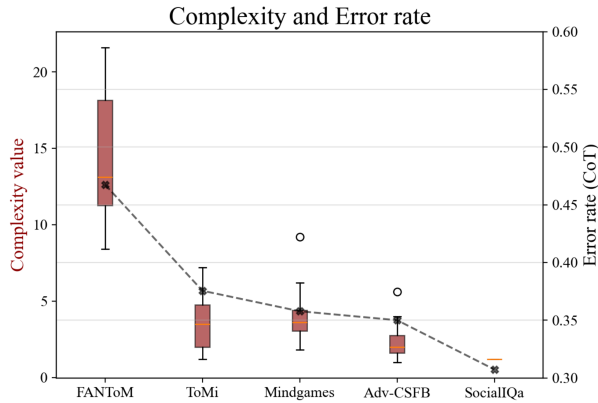


Figure 6: Each boxplot summarizes the complexity analysis of the five ToM benchmarks in ascending order. We report the average error rate (i.e., 1-accuracy) of GPT-3.5-Turbo, GPT-4, Mixtral 8x7B and LLaMA3-70B on the task when prompted with CoT.

and are then queried with the first-order belief of Benjamin. With CoT, the model makes an erroneous assumption about the presence of Benjamin and Isabella in the room. On the other hand, when prompted with DWM, GPT-4 provides an informative description of each *state space*, particularly the knowledge and the uncertainty of each agent’s beliefs, and eventually answers correctly. One example per benchmark is available in the Appendix, Section B.2, while many more are available for inspection in the Code Supplementary Material. Such phenomenon is ubiquitous to all the ToM tasks we tested, a hint that DWM elicits the ToM capabilities of LLMs without requiring external information or solvers.

**Memorisation in Theory of Mind.** Recent works expressed concern about ToM benchmarks’

efficacy in memorisation (Jacovi et al., 2023; La Malfa et al., 2024). This motivated us to quantify and then analyse the impact of memorisation of ToM benchmarks on performance. We computed the percentage of memorised prompts to understand whether that affects the performance of techniques, such as DWM, that split the prompt into chunks and introduce additional information instead of CoT, which leaves the input prompt unchanged. As illustrated in Table 1, ToMi and FANToM have been heavily memorised, with entire portions of the benchmarks that can be retrieved *word for word* from GPT-3.5-Instruct (the auto-complete model by OpenAI). Despite that, no clear evidence of a performance drop in DWM induced by memorisation exists. For GPT-3.5, despite CoT having higher performance on ToMi, DWM is better on FANToM (Figure 4). We hypothesise that as long as a memorised problem is prompted, either in its exact form (as for CoT) or split as in DWM, the most potent models can recover it alongside the ground truth label, thus invalidating the test for both. We conclude with a note of caution. While we discovered that ToMi and FANToM are memorised by GPT-3.5-Instruct, that doesn’t imply any LLM, including GPT-3.5-Turbo and GPT-4, whose training details are not released publicly, has been trained on that data.

## 4.2 Statefulness of ToM Benchmarks

We used the complexity framework introduced in Section 3.1 to characterise the statefulness and statelessness of the five ToM benchmarks used for the experimental evaluation. We randomly sampled 50 problems from each dataset, identified the objects, and manually labelled stateful and stateless

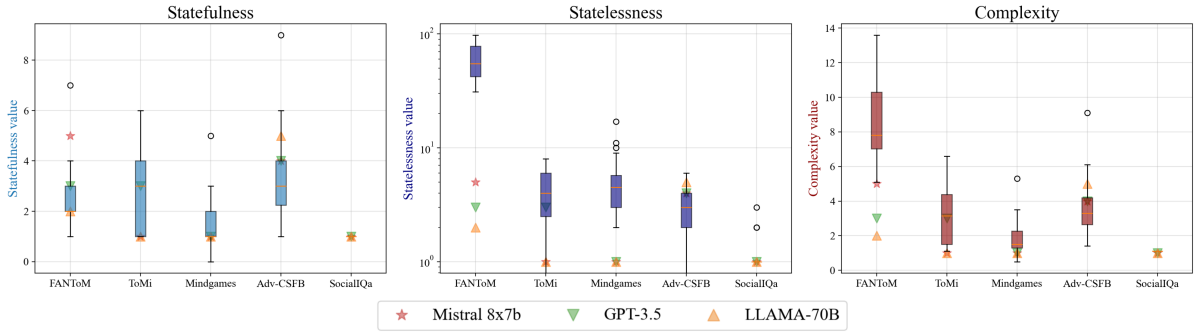


Figure 7: Each boxplot summarizes the statefulness (left), statelessness (middle, y-axis in log-scale) and complexity analysis (right) of the five ToM benchmarks. We report mean, standard deviation and outliers alongside the best DWM method (by the number of prompt splits) and observe a strong correlation between the number of splits and the statefulness.

424 *state events*. We release the split samples alongside  
 425 a web application that facilitates manual labelling.  
 426 As illustrated in Figure 7 (left), the statefulness  
 427 of each problem, i.e., that of the object a model  
 428 must track to answer correctly, **strongly correlates**  
 429 with the best-performing DWM split. The stateless-  
 430 ness complexity, reported in Figure 7 (middle), i.e.,  
 431 that of objects that a model does not need to track,  
 432 grows larger for problems such as FANToM, only  
 433 partially influencing the models’ performance. We  
 434 hypothesise that the most potent models developed  
 435 some competency in discerning the relevant part of  
 436 a prompt (the stateful events) from the confounding  
 437 ones. We finally report, in Figure 7 (right), the  
 438 complexity of each problem computed as per Eq. 2,  
 439 with  $\tau$  set in a range between 0.05 and 0.2 (i.e.,  
 440 the relative weight of stateless compared to stateful  
 441 events). Results suggest that FANToM is the most  
 442 difficult ToM task for humans and LLMs (see Fig-  
 443 ure 4), followed by ToMi (the second most difficult  
 444 for LLMs as well) and Adv-CSFB (which seems  
 445 easier than the others); in contrast, Mindgames and  
 446 SocialIQa tend to be easier. Finally, in Figure 6, we  
 447 compare the accuracy of GPT-3.5-Turbo, GPT-4,  
 448 Mixtral 8x7B and LLaMA3-70B when prompted  
 449 with CoT (i.e., without split) on the five ToM ben-  
 450 chmarks with the complexity of the task as per Def. 2.  
 451 We observe a **strong correlation** between the error-  
 452 rate and the complexity of a task, i.e., our frame-  
 453 work correctly identifies the tasks that are harder  
 454 both for humans and current state-of-the-art LLMs.

## 455 5 Conclusions

456 This paper introduces a complexity framework to  
 457 measure the difficulty of Theory of Mind (ToM)  
 458 problems. It quantifies the difficulty by tracking

459 necessary states (stateful) and unnecessary states  
 460 (stateless), with the latter discounted in the com-  
 461 plexity computation. The framework evidences a  
 462 strong correlation between complexity and model  
 463 performance. Inspired by this framework, we pro-  
 464 pose DWM, a prompting technique that splits a  
 465 prompt into parts to query a model for a consis-  
 466 tent representation of the environment and agents’  
 467 beliefs. DWM outperforms CoT and ToT by ex-  
 468 tracting implicit but relevant information.

## 469 Limitations

470 **Higher order belief tracking.** Our theoretical  
 471 framework reduces the problem of solving a belief  
 472 ToM problem to finding the correct descriptions  
 473 that need to be tracked. It extends seamlessly to  
 474 tasks with much higher complexity, however, we  
 475 have not had the opportunity to test this theory in  
 476 those settings. We noticed that most theory of mind  
 477 tasks available in the community only require one  
 478 to five states to be correctly answered. A possible  
 479 extension would be testing the theory upon tasks  
 480 with higher state complexity, e.g.  $k^{\text{th}}$ -order belief  
 481 tracking tasks. However, it is unclear whether this  
 482 could be useful in real applications as most human  
 483 belief tracking is limited to 5 or 6 orders (Cargile,  
 484 1970; Dennett, 1988).

485 **On task splitting methods.** It is not straightfor-  
 486 ward to automatically find the correct task splits  
 487 in a manner that correctly describes the state. An  
 488 LLM could find a way to split it by itself correctly  
 489 and use those splits to answer the question. We at-  
 490 tempted this approach, yet with a simple prompting  
 491 method, the model splits every sentence, making  
 492 the descriptions much noisier and less accurate.  
 493 Future work could try to find the best splits auto-



494	atically.		
495	<b>Memorization analysis.</b> Training and evaluating		
496	on the same dataset produce positively biased data		
497	on the model’s performance. While running our		
498	benchmarks on ToMi, we discovered that the GPT-		
499	3.5 model had completely memorized parts of the		
500	dataset. This motivated us to extend the memoriza-		
501	tion test to the other tasks. We urge the research		
502	community to include a memorization section on		
503	every benchmark study with public datasets used		
504	in their works. This data is crucial to conduct fair		
505	and unbiased research on evaluating LLMs’ abili-		
506	ties (Jacovi et al., 2023). Future works will include		
507	an analysis of the memorisation rate of other ToM		
508	tasks alongside tests to quantify their impact on		
509	different models.		
510	<b>Ethical Statement</b>		
511	The datasets and pre-trained LLMs that we use		
512	are all publicly available. This paper focuses on		
513	ToM problems’ hardness and prompting methods.		
514	We highlight that LLMs do not guarantee the pro-		
515	duction of factual data or correct reasoning steps,		
516	and the prompting methods developed here should		
517	not be regarded as the source of truth in making		
518	decisions.		
519	<b>References</b>		
520	AI@Meta. 2024. <a href="#">Llama 3 model card</a> .		
521	Jaan Aru, Aqeel Labash, Oriol Corcoll, and Raul Vi-		
522	cente. 2023. Mind the gap: Challenges of deep learn-		
523	ing approaches to theory of mind. <i>Artificial Intelli-</i>		
524	<i>gence Review</i> , 56(9):9141–9156.		
525	Chris Baker, Rebecca Saxe, and Joshua Tenenbaum.		
526	2011. <a href="#">Bayesian Theory of Mind: Modeling Joint</a>		
527	<a href="#">Belief-Desire Attribution</a> .		
528	Simon Baron-Cohen, Alan M. Leslie, and Uta Frith.		
529	1985. <a href="#">Does the autistic child have a “theory of mind”</a>		
530	? 21(1):37–46.		
531	Sébastien Bubeck, Varun Chandrasekaran, Ronen El-		
532	dan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Pe-		
533	ter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg,		
534	Harsha Nori, Hamid Palangi, Marco Tulio Ribeiro,		
535	and Yi Zhang. 2023. <a href="#">Sparks of Artificial General</a>		
536	<a href="#">Intelligence: Early experiments with GPT-4</a> .		
537	James Cargile. 1970. <a href="#">A note on" iterated knowings"</a> .		
538	<i>Analysis</i> , 30(5):151–155.		
539	Zhuang Chen, Jincenzi Wu, Jinfeng Zhou, Bosi Wen,		
540	Guanqun Bi, Gongyao Jiang, Yaru Cao, Mengt-		
541	ing Hu, Yunghwei Lai, Zexuan Xiong, et al. 2024.		
	Tombench: Benchmarking theory of mind in large	542	
	language models. <i>arXiv preprint arXiv:2402.15052</i> .	543	
	Paul M Churchland. 2013. Folk psychology and the	544	
	explanation of human behavior 1. In <i>Folk psychology</i>	545	
	<i>and the philosophy of mind</i> , pages 247–262. Psychol-	546	
	ogy Press.	547	
	Daniel C. Dennett. 1988. The intentional stance in	548	
	theory and practice. In <i>Machiavellian Intelligence:</i>	549	
	<i>Social Expertise and the Evolution of Intellect in</i>	550	
	<i>Monkeys, Apes, and Humans</i> , pages 180–202. Claren-	551	
	don Press/Oxford University Press.	552	
	Kanishk Gandhi, Gala Stojnic, Brenden M. Lake, and	553	
	Moira R. Dillon. 2022. <a href="#">Baby Intuitions Benchmark</a>	554	
	<a href="#">(BIB): Discerning the goals, preferences, and actions</a>	555	
	<a href="#">of others</a> .	556	
	Alison Gopnik and Henry M. Wellman. 1994. <a href="#">The</a>	557	
	<a href="#">theory theory</a> . In Lawrence A. Hirschfeld and Su-	558	
	san A. Gelman, editors, <i>Mapping the Mind: Domain</i>	559	
	<i>Specificity in Cognition and Culture</i> , pages 257–293.	560	
	Cambridge University Press.	561	
	Sven Gronauer and Klaus Diepold. 2022. Multi-agent	562	
	deep reinforcement learning: a survey. <i>Artificial</i>	563	
	<i>Intelligence Review</i> , 55(2):895–943.	564	
	David Ha and Jürgen Schmidhuber. 2018. <a href="#">World Mod-</a>	565	
	<a href="#">els</a> .	566	
	Shibo Hao, Yi Gu, Haodi Ma, Joshua Hong, Zhen	567	
	Wang, Daisy Wang, and Zhiting Hu. 2023. <a href="#">Reason-</a>	568	
	<a href="#">ing with language model is planning with world</a>	569	
	<a href="#">model</a> . In <i>Proceedings of the 2023 Conference on</i>	570	
	<i>Empirical Methods in Natural Language Processing</i> ,	571	
	pages 8154–8173, Singapore. Association for Com-	572	
	putational Linguistics.	573	
	Alon Jacovi, Avi Caciularu, Omer Goldman, and Yoav	574	
	Goldberg. 2023. <a href="#">Stop uploading test data in plain</a>	575	
	<a href="#">text: Practical strategies for mitigating data contam-</a>	576	
	<a href="#">ination by evaluation benchmarks</a> . <i>ArXiv preprint</i> ,	577	
	abs/2305.10160.	578	
	Albert Q Jiang, Alexandre Sablayrolles, Antoine	579	
	Roux, Arthur Mensch, Blanche Savary, Chris Bam-	580	
	ford, Devendra Singh Chaplot, Diego de las Casas,	581	
	Emma Bou Hanna, Florian Bressand, et al. 2024.	582	
	Mixtral of experts. <i>arXiv preprint arXiv:2401.04088</i> .	583	
	Hyunwoo Kim, Melanie Sclar, Xuhui Zhou, Ronan Le	584	
	Bras, Gunhee Kim, Yejin Choi, and Maarten Sap.	585	
	2023. <a href="#">FANToM: A Benchmark for Stress-testing</a>	586	
	<a href="#">Machine Theory of Mind in Interactions</a> .	587	
	Najoung Kim and Sebastian Schuster. 2023. <a href="#">En-</a>	588	
	<a href="#">tity tracking in language models</a> . <i>Preprint</i> ,	589	
	arXiv:2305.02363.	590	
	Najoung Kim, Sebastian Schuster, and Shubham Tosh-	591	
	niwal. 2024. <a href="#">Code pretraining improves entity</a>	592	
	<a href="#">tracking abilities of language models</a> . <i>Preprint</i> ,	593	
	arXiv:2405.21068.	594	

595	Michal Kosinski. 2023. Theory of mind may have spontaneously emerged in large language models. <i>arXiv preprint arXiv:2302.02083</i> , 4:169.	648
596		649
597		650
598	Emanuele La Malfa, Aleksandar Petrov, Simon Frieder, Christoph Weinhuber, Ryan Burnell, Raza Nazar, Anthony G. Cohn, Nigel Shadbolt, and Michael Wooldridge. 2023. <a href="#">Language Models as a Service: Overview of a New Paradigm and its Challenges</a> .	651
599		652
600		653
601		
602		
603	Emanuele La Malfa, Christoph Weinhuber, Orazio Torre, Fangru Lin, Anthony Cohn, Nigel Shadbolt, and Michael Wooldridge. 2024. <a href="#">Code Simulation Challenges for Large Language Models</a> .	654
604		655
605		656
606		
607	Matthew Le, Y-Lan Boureau, and Maximilian Nickel. 2019. <a href="#">Revisiting the Evaluation of Theory of Mind through Question Answering</a> . In <i>Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)</i> , pages 4463–4473. Association for Computational Linguistics.	657
608		658
609		659
610		660
611		661
612		662
613		663
614		664
615	Belinda Z. Li, Maxwell Nye, and Jacob Andreas. 2021. <a href="#">Implicit representations of meaning in neural language models</a> . <i>Preprint</i> , arXiv:2106.00737.	665
616		666
617		
618	Kenneth Li, Aspen K Hopkins, David Bau, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2022. Emergent world representations: Exploring a sequence model trained on a synthetic task. <i>arXiv preprint arXiv:2210.13382</i> .	667
619		668
620		669
621		670
622		671
623	Kyle Mahowald, Anna A Ivanova, Idan A Blank, Nancy Kanwisher, Joshua B Tenenbaum, and Evelina Fedorenko. 2024. Dissociating language and thought in large language models. <i>Trends in Cognitive Sciences</i> .	672
624		673
625		674
626		675
627	John McCarthy. 1979. Ascribing Mental Qualities To Machines.	676
628		677
629	Shima Rahimi Moghaddam and Christopher J. Honey. 2023. <a href="#">Boosting theory-of-mind performance in large language models via prompting</a> . <i>Preprint</i> , arXiv:2304.11490.	678
630		679
631		680
632		681
633	Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. 2021. Show your work: Scratchpads for intermediate computation with language models. <i>arXiv preprint arXiv:2112.00114</i> .	682
634		683
635		684
636		685
637		686
638		687
639	OpenAI. 2023. <a href="#">GPT-4 technical report</a> . <i>ArXiv preprint</i> , abs/2303.08774.	688
640		689
641	Joon Sung Park, Joseph C. O’Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. <a href="#">Generative agents: Interactive simulacra of human behavior</a> . <i>Preprint</i> , arXiv:2304.03442.	690
642		691
643		692
644		693
645		
646	David Premack and Guy Woodruff. 1978. Does the chimpanzee have a theory of mind? 1(4):515–526.	694
647		695
	Stephanie D Preston and Frans BM De Waal. 2002. Empathy: Its ultimate and proximate bases. <i>Behavioral and brain sciences</i> , 25(1):1–20.	696
		697
		698
		699
	Neil C. Rabinowitz, Frank Perbet, H. Francis Song, Chiyuan Zhang, S. M. Ali Eslami, and Matthew Botvinick. 2018. <a href="#">Machine Theory of Mind</a> .	
	Maarten Sap, Ronan LeBras, Daniel Fried, and Yejin Choi. 2023. <a href="#">Neural Theory-of-Mind? On the Limits of Social Intelligence in Large LMs</a> .	
	Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019. <a href="#">Social IQa: Commonsense Reasoning about Social Interactions</a> . In <i>Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)</i> , pages 4463–4473. Association for Computational Linguistics.	
	Brian Scassellati. 2002. <a href="#">Theory of Mind for a Humanoid Robot</a> . 12(1):13–24.	
	Natalie Shapira, Mosh Levy, Seyed Hossein Alavi, Xuhui Zhou, Yejin Choi, Yoav Goldberg, Maarten Sap, and Vered Shwartz. 2023. <a href="#">Clever Hans or Neural Theory of Mind? Stress Testing Social Reasoning in Large Language Models</a> .	
	Damien Sileo and Antoine Lerneuld. 2023. <a href="#">MindGames: Targeting Theory of Mind in Large Language Models with Dynamic Epistemic Modal Logic</a> .	
	James WA Strachan, Dalila Albergo, Giulia Borghini, Oriana Pansardi, Eugenio Scaliti, Saurabh Gupta, Krati Saxena, Alessandro Rufo, Stefano Panzeri, Guido Manzi, et al. 2024. Testing theory of mind in large language models and humans. <i>Nature Human Behaviour</i> , pages 1–11.	
	Michael Tomasello. 2009. <i>The cultural origins of human cognition</i> . Harvard university press.	
	Shubham Toshniwal, Sam Wiseman, Karen Livescu, and Kevin Gimpel. 2021. <a href="#">Learning chess blindfolded: Evaluating language models on state tracking</a> . <i>CoRR</i> , abs/2102.13249.	
	Tomer Ullman. 2023. <a href="#">Large Language Models Fail on Trivial Alterations to Theory-of-Mind Tasks</a> .	
	Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. <a href="#">Chain-of-Thought Prompting Elicits Reasoning in Large Language Models</a> .	
	Henry M. Wellman. 2017. <a href="#">The Development of Theory of Mind: Historical Reflections</a> . 11(3):207–214.	
	Heinz Wimmer and Josef Perner. 1983. Beliefs about beliefs: Representation and constraining function of wrong beliefs in young children’s understanding of deception. 13(1):103–128.	

700 Lionel Wong, Gabriel Grand, Alexander K. Lew,  
701 Noah D. Goodman, Vikash K. Mansinghka, Jacob  
702 Andreas, and Joshua B. Tenenbaum. 2023. [From](#)  
703 [Word Models to World Models: Translating from](#)  
704 [Natural Language to the Probabilistic Language of](#)  
705 [Thought](#).

706 Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran,  
707 Thomas L. Griffiths, Yuan Cao, and Karthik  
708 Narasimhan. 2023. [Tree of Thoughts: Deliberate](#)  
709 [Problem Solving with Large Language Models](#).

710 Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei,  
711 Nathan Scales, Xuezhi Wang, Dale Schuurmans,  
712 Claire Cui, Olivier Bousquet, Quoc V Le, and Ed H.  
713 Chi. 2023a. [Least-to-most prompting enables com-](#)  
714 [plex reasoning in large language models](#). In *The*  
715 *Eleventh International Conference on Learning Rep-*  
716 *resentations*.

717 Pei Zhou, Aman Madaan, Srividya Pranavi Potharaju,  
718 Aditya Gupta, Kevin R. McKee, Ari Holtzman,  
719 Jay Pujara, Xiang Ren, Swaroop Mishra, Aida Ne-  
720 matzadeh, Shyam Upadhyay, and Manaal Faruqi.  
721 2023b. [How far are large language models from](#)  
722 [agents with theory-of-mind?](#)

## A Experimental Setup

### A.1 Experimental Details

Most of the language models used in this work follow the Language Models as a Service (LMaaS) paradigm (La Malfa et al., 2023). This model of service does not allow transparency and hinders reproducibility. Reproducibility is difficult to achieve as common software development frameworks, such as CI/CD pipeline, ease the update of the public service, but change the underlying entity. From this follows that the model tested by the researcher could change at any time. This is not solvable from the outside. Researchers have no control over the software engineering practices inside a LMaaS, but could set some parameters to offer the highest possible grade of reproducibility. We set the temperature to zero or enable greedy decoding by default (this does not imply determinism even if model weights are not changed).<sup>2</sup> In prompting methods where the creativity of the response is exploited for better performance, e.g., Tree of Thoughts (Yao et al., 2023), we set the temperature to 0.7, the value proposed in the reference papers.

**LMaaS providers.** We use Huggingface for Mixtral 8x7B. Groq Cloud for LLama-3-7B and LLama-3-70B. Microsoft sponsorship for GPT-3.5 and GPT-4 access.

### A.2 Prompting Templates

We present the different prompting techniques, taking as an example the following prompt from ToMi and GPT-3.5-Turbo as the reference model:

- 
1. Benjamin entered the workshop.
  2. Isabella entered the workshop.
  3. Hannah entered the workshop.
  4. Isabella hates the onion
  5. Hannah hates the t-shirt
  6. The pajamas is in the bottle.
  7. Isabella moved the pajamas to the drawer.
  8. Benjamin exited the workshop.
  9. Isabella exited the workshop.
  10. Benjamin entered the workshop.
- 

And the following question:

---

Where does Benjamin think that Isabella searches for the pajamas?

---

### Chain of Thought

<sup>2</sup>The main explanation are the <https://github.com/pytorch/pytorch/issues/75240>"non deterministic cuda cores" another could "be batched inference in sparse MoE models", see <https://152334h.github.io/blog/non-determinism-in-gpt-4/here>

---

Consider the following dialogue where multiple agents interact. At the end, I will ask you a question to answer.

Here's the dialogue:

1. Benjamin entered the workshop.
2. Isabella entered the workshop.
3. Hannah entered the workshop.
4. Isabella hates the onion
5. Hannah hates the t-shirt
6. The pajamas is in the bottle.
7. Isabella moved the pajamas to the drawer.
8. Benjamin exited the workshop.
9. Isabella exited the workshop.
10. Benjamin entered the workshop.

This is the end of the dialogue. Now, this is a question for you to answer.

Question: Where does Benjamin think that Isabella searches for the pajamas?

Think step by step, answer the question with one word and provide the answer between <answer></answer> tags.

For example, reply with <answer>vase</answer>.

---

### Tree of Thought

We first prompt an LLM to propose different solution paths to solve a task.

---

Consider the following dialogue where multiple agents interact. At the end, I will ask you a question to answer.

Here's the dialogue:

1. Benjamin entered the workshop.
2. Isabella entered the workshop.
3. Hannah entered the workshop.
4. Isabella hates the onion
5. Hannah hates the t-shirt
6. The pajamas is in the bottle.
7. Isabella moved the pajamas to the drawer.
8. Benjamin exited the workshop.
9. Isabella exited the workshop.
10. Benjamin entered the workshop.

Question: Where does Benjamin think that Isabella searches for the pajamas?

Think step by step and list all possible answers providing a single answer on each line.

---

We then pick the best idea via a majority vote over three agents simulated by the LLM itself:

---

Given a dialogue and several observation choices, decide which choice is most promising. Analyze each choice in detail, then conclude in the last line "The best choice is {{s}}", where s the integer id of the choice.

1. Benjamin entered the workshop.
2. Isabella entered the workshop.
3. Hannah entered the workshop.
4. Isabella hates the onion

840 5. Hannah hates the t-shirt  
 841 6. The pajamas is in the bottle.  
 842 7. Isabella moved the pajamas to the drawer.  
 843 8. Benjamin exited the workshop.  
 844 9. Isabella exited the workshop.  
 845 10. Benjamin entered the workshop.  
 846  
 847 Here are some possible observations:  
 848 **## Here we insert the output of the previous**  
 849 **prompt.**

---

851 **We eventually ask the model for a final answer.**

---

852 Given this dialogue and possible observations,  
 853 answer the question with one word and  
 854 provide the answer between  
 855 <answer></answer> tags.  
 856  
 857 1. Benjamin entered the workshop.  
 858 2. Isabella entered the workshop.  
 859 3. Hannah entered the workshop.  
 860 4. Isabella hates the onion  
 861 5. Hannah hates the t-shirt  
 862 6. The pajamas is in the bottle.  
 863 7. Isabella moved the pajamas to the drawer.  
 864 8. Benjamin exited the workshop.  
 865 9. Isabella exited the workshop.  
 866 10. Benjamin entered the workshop.  
 867  
 868 Question: Where does Benjamin think that  
 869 Isabella searches for the pajamas?  
 870  
 871 **## Here we insert the observations generated by**  
 872 **the LLM with the previous prompts.**  
 873  
 874 For example, reply with <answer>vase</answer>.

---

876 **Discrete World Models - 1 Split**

---

877 I give you a phrase of a dialogue between  
 878 agents. I will reveal more parts of it  
 879 later. At the end, I will give you a  
 880 question you must answer.  
 881 For each phrase, you must:  
 882 # 1. Write down a succinct description of what  
 883 each agent knows about the environment and  
 884 about the other agents. Keep the  
 885 description short and do not produce  
 886 redundant information.  
 887 # 2. Each considerations you make must be  
 888 preceded by the symbol #GPT#.  
 889 Here's the dialogue:  
 890  
 891 1. Benjamin entered the workshop.  
 892 2. Isabella entered the workshop.  
 893 3. Hannah entered the workshop.  
 894 4. Isabella hates the onion  
 895 5. Hannah hates the t-shirt  
 896 6. The pajamas is in the bottle.  
 897 7. Isabella moved the pajamas to the drawer.  
 898 8. Benjamin exited the workshop.  
 899 9. Isabella exited the workshop.  
 900 10. Benjamin entered the workshop.  
 901  
 902 This is the end of the dialogue. Now, this is a  
 903 question for you to answer.  
 904  
 905 Question: Where does Benjamin think that  
 906 Isabella searches for the pajamas?  
 907  
 908

Think step by step, answer the question with  
 one word and provide the answer between  
 <answer></answer> tags.  
 For example, reply with <answer>vase</answer>.

---

**Discrete World Model - 3 Split**

---

I give you a phrase of a dialogue between  
 agents. I will reveal more parts of it  
 later. At the end, I will give you a  
 question you must answer.  
 For each phrase, you must:  
 # 1. Write down a succinct description of what  
 each agent knows about the environment and  
 about the other agents. Keep the  
 description short and do not produce  
 redundant information.  
 # 2. Each considerations you make must be  
 preceded by the symbol #GPT#.  
 Here's the dialogue:  
 1. Benjamin entered the workshop.  
 2. Isabella entered the workshop.  
 3. Hannah entered the workshop.  
**## Here the LLM provides a description of the**  
**environment so far described by the**  
**dialogue.**  
 4. Isabella hates the onion  
 5. Hannah hates the t-shirt  
 6. The pajamas is in the bottle.  
**## Here the LLM provides a description of the**  
**environment so far described by the**  
**dialogue.**  
 7. Isabella moved the pajamas to the drawer.  
 8. Benjamin exited the workshop.  
 9. Isabella exited the workshop.  
 10. Benjamin entered the workshop.  
 This is the end of the dialogue. Now, this is a  
 question for you to answer.  
 Question: Where does Benjamin think that  
 Isabella searches for the pajamas?  
 Think step by step, answer the question with  
 one word and provide the answer between  
 <answer></answer> tags.  
 For example, reply with <answer>vase</answer>.

---

**Yaml/JSON**

---

Consider the following dialogue where multiple  
 agents interact.

1. Benjamin entered the workshop.  
 2. Isabella entered the workshop.  
 3. Hannah entered the workshop.  
 4. Isabella hates the onion  
 5. Hannah hates the t-shirt  
 6. The pajamas is in the bottle.  
 7. Isabella moved the pajamas to the drawer.  
 8. Benjamin exited the workshop.  
 9. Isabella exited the workshop.  
 10. Benjamin entered the workshop.

Here is the YAML representation of the text.  
**## Here we substitute the JSON/Yaml**  
**representation of the dialogue (see next**

979 **prompt**).

980  
981 Question: Question: Where does Benjamin think  
982 that Isabella searches for the pajamas?

983  
984 Answer between the tags with a single word that  
985 is the answer of the above question  
986 For example <answer>vase</answer>.

---

988 The JSON/YAML representation is required  
989 with the following prompt:

---

990 Consider the following dialogue where multiple  
991 agents interact.

- 992  
993 1. Benjamin entered the workshop.  
994 2. Isabella entered the workshop.  
995 3. Hannah entered the workshop.  
996 4. Isabella hates the onion  
997 5. Hannah hates the t-shirt  
998 6. The pajamas is in the bottle.  
999 7. Isabella moved the pajamas to the drawer.  
1000 8. Benjamin exited the workshop.  
1001 9. Isabella exited the workshop.  
1002 10. Benjamin entered the workshop.

1003  
1004 Now give a structured representation of the  
1005 dialogue in YAML format. Keep track of the  
1006 information that each agent has access to  
1007 at each point in the dialogue.

1008 It is important to have a relative  
1009 representation of the information that each  
1010 agent has access to at each point in the  
1011 dialogue.

---

## 1013 **B Additional Results**

### 1014 **B.1 DWM Prompting**

1015 In this section, and, in particular in Figure 8, we  
1016 report results for LLaMA3-7B, LLaMA3-70B and  
1017 GPT-4 on the five ToM benchmarks and for differ-  
1018 ent prompting techniques, namely DWM (one to  
1019 five splits), JSON, Yaml, CoT and ToT.

### 1020 **B.2 DWM Elicits More Informed Mental** 1021 **States in LLMs**

1022 In this section, we report and discuss an example of  
1023 a real prompt and the answers provided by GPT-4  
1024 for each ToM task we evaluated in this paper. For  
1025 FANToM (Figure 10), we just reported the portion  
1026 of the prompt that induces an unfaithful reasoning  
1027 process in GPT-4, due to the prohibitive length of  
1028 the input prompts. Results for ToMi, FANToM,  
1029 ADV-csfb, Mindgames and SocialIQa are reported  
1030 respectively in Figures 9, 10, 11, 12 and 13.

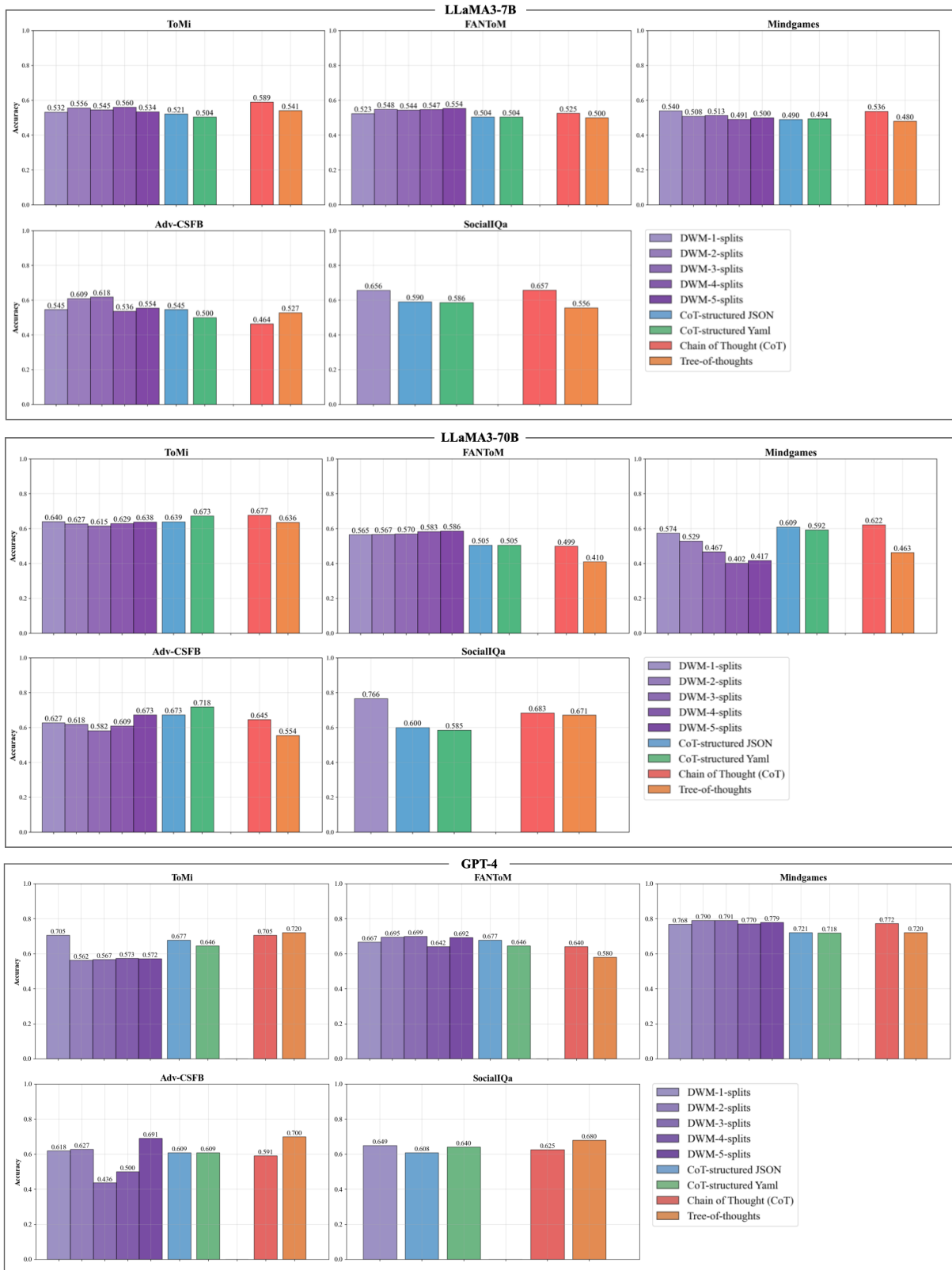


Figure 8: Benchmarks of LLaMA3-7B (top), LLaMA3-70B (middle) and GPT-4 (bottom) models on different ToM tasks for DWM (one to five splits), CoT, ToT and structured prompts (JSON and Yaml). For GPT-4 and ToT, we tested 50 samples (instead of 1000).

### Input Prompt

Benjamin entered the workshop.  
Isabella entered the workshop.  
Hannah entered the workshop.  
Isabella hates the onion  
Hannah hates the t-shirt  
The pajamas is in the bottle.  
Isabella moved the pajamas to the drawer.  
Benjamin exited the workshop.  
Isabella exited the workshop.  
Benjamin entered the workshop.

Where does Benjamin think that Isabella searches for the pajamas?

### Question

### Chain of Thoughts

“The pajamas are initially in the bottle. [...] **Benjamin’s Presence: Benjamin is not present in the workshop when Isabella moves the pajamas.** He exits before this action and re-enters after Isabella has already left. Based on this sequence of events, **Benjamin was not present to witness Isabella moving the pajamas to the drawer.** Therefore, in Benjamin’s understanding, the last known location of the pajamas would still be where they were originally placed before he left, which is the bottle. <answer>bottle</answer>” ✗

“**Benjamin knows** that he entered the workshop, that Isabella and Hannah also entered, that Isabella hates onion and Hannah hates t-shirts, **that the pajamas were in the bottle but Isabella moved them to the drawer, and that he exited and re-entered the workshop. He doesn’t know that Isabella exited the workshop.** [...] <answer>drawer</answer>” ✓

### Discrete World Models

Figure 9: Example of a real ToMi example where GPT-4 fails when prompted with CoT, yet succeeds with DWM. CoT elicits an untruthful reasoning process (in red), while DWM correctly informs the model with the correct information about Benjamin’s first-order belief (in green).

### Input Prompt

[...]  
**Conor:** Raegan, you were saying about being a vegetarian. Has it had a big impact on your personal life?  
**Raegan:** Absolutely, Conor. It's definitely been a transformation. Initially, it was challenging to find vegetarian alternatives to my favorite meals, but over time, it's made me be more aware of what I'm eating. It's also made me try out a lot of new and interesting foods, vegan ones included!  
**Conor:** That's really insightful. I've been considering it, mainly for health reasons, but haven't taken the leap yet. Maybe trying out that vegan restaurant is a good place to start.  
**Raegan:** Definitely! Exploring vegan or vegetarian food can be a great first step. The variety and flavors might surprise you!  
**Conor:** I'll keep that in mind. It's always exciting to discover something new!  
**Brittney:** Hey guys, I'm back! I find our talks about food trends so interesting.  
[...]

### Question

What does Brittney believe were Conor's primary motivations for considering a shift to vegetarianism?

### Chain of Thoughts

“Based on the dialogue, Brittney did not explicitly state what she believes Conor's primary motivations are for considering a shift to vegetarianism. However, **Conor himself mentions that he has been considering vegetarianism mainly for health reasons. Thus, if Brittney is aware of this part of the conversation,** she would likely think that health reasons are Conor's primary motivation for considering the change to a vegetarian diet. <answer>Health reasons</answer>” ✗

### Discrete World Models

“Brittney knows that she was part of a conversation with Conor and Raegan. **She had to leave temporarily due to her parking meter expiring.** She also knows that the conversation was about food trends, specifically vegetarianism and veganism, and finds it interesting [...] **She also knows that Conor is considering becoming a vegetarian** and has suggested trying out vegan or vegetarian food as a first step. [...] <answer>brittney doesn't have information about conor's primary motivations for considering a shift to vegetarianism as she was not present during that part of the conversation.</answer>” ✓

Figure 10: Example of a real FANToM example where GPT-4 fails when prompted with CoT, yet succeeds with DWM. CoT elicits an untruthful reasoning process (in red), while DWM correctly informs the model with the correct information about the partial observability Brittney has about Conor (in green).



### Input Prompt

There is a closed glass box on the table of an empty meeting room, in which a seminar is about to take place.

### Question

Fill in the blank with the best option. She loves writing with \_  
- pens  
- pencils

### Chain of Thoughts

“Based on the information provided, **the attendee reads the label on the box marked "pencils" and feels delighted upon finding it. This suggests that she has a preference for what she believes the box contains, which according to the label are pencils, even though the box actually contains pens.** Therefore, the best answer, given her reaction to the label, would be:  
<answer>**pencils**</answer>” ✘

### Discrete World Models

“Agent: James  
**James knows that the box on the table is filled with pens, not pencils, despite the label saying "pencils".** He knows that he labeled the box himself. He is aware of his own actions and the contents of the box. He also knows that he is alone in the room and that a seminar is about to take place.  
<answer>**pens**</answer>” ✔

Figure 11: Example of a real ADV-csfb example where GPT-4 fails when prompted with CoT, yet succeeds with DWM. CoT elicits an untruthful reasoning process (in **red**), while DWM correctly informs the model with the correct information about the content of the glass box (in **green**).

### Input Prompt

There are three persons. Everyone is visible to others. Each person draws a card, face unrevealed (red or black). Leah's card is shown to Leah. Raymond's card is shown to Leah. Brittney's card is shown to Leah. Raymond's card is shown to Raymond. It is publicly announced that someone picked a red card.

### Question

Leah can now know that Raymond can know whether or not nobody picked a red card.  
Is the above statement in entailment or contradiction with the previous situation?

### Chain of Thoughts

“In the given situation, Leah sees Raymond's card and Brittney's card, but she does not see her own card. It is publicly announced that someone picked a red card, but **Leah cannot determine whether it was Raymond or Brittney who picked the red card based on the information she has.**  
<answer>**contradiction**</answer> ❌

### Discrete World Models

“Leah: Knows the color of her own card. Does not know the color of the other two persons' cards. Knows that the other two persons can see her.  
Other two persons: Do not know the color of their own cards or each other's cards. Knows that Leah knows the color of her card. Knows that Leah can see them.  
Leah: Knows the color of her own card, Raymond's card, and Brittney's card. **Knows that Raymond knows the color of his card.** Knows that someone picked a red card. Raymond: Knows the color of his own card. Does not know the color of Leah's or Brittney's cards. Knows that Leah knows the color of his card and Brittney's card. Knows that someone picked a red card.  
Brittney: Does not know the color of her own card or Raymond's card. Knows that Leah knows the color of her card and Raymond's card. Knows that Raymond knows the color of his card. Knows that someone picked a red card. <answer>**entailment**</answer>” ✓

Figure 12: Example of a real Mindgames example where GPT-4 fails when prompted with CoT, yet succeeds with DWM. CoT elicits an untruthful reasoning process (in **red**), while DWM correctly informs the model with the correct information about the knowledge Leah has about Raymond (in **green**).

### Input Prompt

Skylar was pushed by the bully in the lunch line, so Skylar poured milk over their head.

### Question

Consider the following statement, you will need to infer the correct response. At the end, I will ask you a question to answer.

Here's the statement:

Skylar was pushed by the bully in the lunch line, so Skylar poured milk over their head.

This is the end of the dialogue. Now, this is a question for you to answer.

Question: What will Skylar want to do next?

1: Run away

2: Wait for a reaction

3: wanted to fight with the bully

### Chain of Thoughts

“Based on the statement, Skylar's response to being pushed by the bully was to pour milk over their head. **This suggests that Skylar wanted to retaliate or stand up to the bully**, indicating a desire to fight back.

<answer>**3: wanted to fight with the bully**</answer> ✘

### Discrete World Models

“Agent: Skylar Knowledge about the environment: Skylar knows they are in a lunch line and that there is a bully present. Skylar also knows they have access to milk. Knowledge about other agents: **Skylar knows the bully is aggressive and has pushed them.**

Agent: The Bully Knowledge about the environment: The bully knows they are in a lunch line and that Skylar is present. Knowledge about other agents: The bully knows Skylar is someone they can push around, but may not anticipate Skylar's retaliatory action.

<answer>**1: Run away**</answer>” ✔

Figure 13: Example of a real SocialIQa example where GPT-4 fails when prompted with CoT, yet succeeds with DWM. CoT elicits an untruthful reasoning process (in **red**), while DWM correctly informs the model with the correct next action Skylar will take (in **green**).