GuessingGame: Measuring the Informativeness of Open-Ended Questions in Large Language Models

Dylan Hutson, Daniel Vennemeyer, Aneesh Deshmukh, Justin Zhan, and Tianyu Jiang
University of Cincinnati
{hutsondm, vennemdp, deshmua2}@mail.uc.edu, zhanjt@ucmail.uc.edu
tianyu.jiang@uc.edu

Abstract

We introduce GuessingGame, a protocol for evaluating large language models (LLMs) as strategic question-askers in open-ended, open-domain settings. A Guesser LLM identifies a hidden object by posing free-form questions to an Oracle without predefined choices or candidate lists. To measure question quality, we propose two information gain (IG) metrics: a Bayesian method that tracks belief updates over semantic concepts using LLM-scored relevance, and an entropy-based method that filters candidates via ConceptNet. Both metrics are model-agnostic and support post hoc analysis. Across 858 games with multiple models and prompting strategies, higher IG strongly predicts efficiency: a one-standard-deviation IG increase reduces expected game length by 43%. Prompting constraints guided by IG, such as enforcing question diversity, enable weaker models to significantly improve performance. These results show that question-asking in LLMs is both measurable and improvable, and crucial for interactive reasoning.

1 Introduction

Large language models (LLMs) excel at factual recall, arithmetic reasoning, and multi-turn dialogue (Brown et al., 2020; OpenAI et al., 2024a). However, while their performance as *answerers* is well studied, their capacity as *askers*, formulating strategic, adaptive, and information-seeking questions, remains less explored. This limitation matters in interactive applications such as education (Chen et al., 2024a), medical diagnosis (Li et al., 2024), and autonomous decision making (Wang et al., 2024), where effective question generation is the key to identifying knowledge gaps and eliciting relevant information—where knowing what to ask can matter more than knowing how to answer.

Despite their fluency, LLMs often ask vague or redundant questions (Mazzaccara et al., 2024). Few

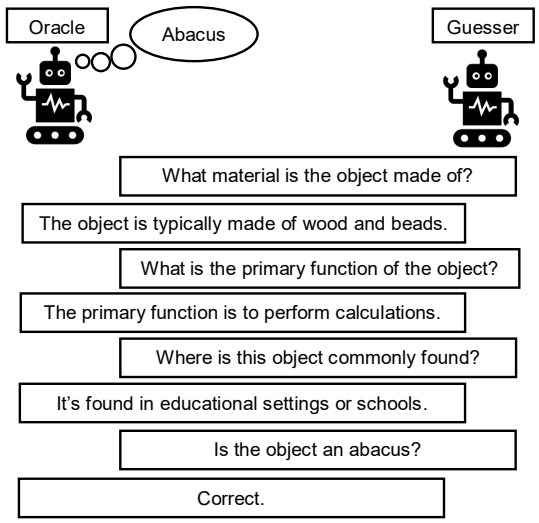

Figure 1: Example of a GuessingGame interaction: the Guesser identifies an abacus through open-ended questions.

standardized protocols exist to evaluate question-asking strategies in unconstrained open-domain settings. Most limit queries to yes/no format (Bertolazzi et al., 2023), constrain the hypothesis space (Aliannejadi et al., 2019), or assume a fully known planning context (Zhang et al., 2024). As a result, we lack a robust way to evaluate how LLMs generate purposeful, informative questions in unconstrained, real-world settings.

We address this gap with **GuessingGame**, an evaluation protocol in which a *Guesser* LLM identifies a hidden object by asking free-form questions to an *Oracle* LLM (Figure 1). The setting is fully open-domain (no candidate list is provided) and open-ended (questions may take any form, not just binary). To analyze behavior, we define a five-part taxonomy of question types: Attribute, Function, Location, Category, and Direct guesses, and measure performance by *success rate* and *average number of questions* to reach the answer.

One disadvantage of these two metrics is that

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 17345–17361
November 4-9, 2025

they only provide useful information for successful guesses—for instance, the number of questions is always fixed at the maximum limit when a game fails. To address this, we propose two information gain (IG) measures estimating uncertainty reduction per question. The first is a Bayesian belief-tracking metric that uses LLM-generated relevance scores to update a distribution over semantic concepts. The second is a ConceptNet (Speer et al., 2017) based metric that filters candidate objects using knowledge graph assertions implied by each question and its answer, estimating IG as the reduction in entropy over the object set. These metrics allow us to quantify question informativeness without requiring access to model internals or ground-truth beliefs.

We evaluate our framework across 858 games, testing a range of prompting strategies and model families. We find that open-ended prompts consistently outperform binary (yes/no) questions, improving success from 32.1% to 39.4% with LLaMA-3.3 70B (Grattafiori et al., 2024). Attribute-based questions (e.g., about size, material, or shape) emerge as the most informative, achieving the highest average IG and the best task performance when used in isolation. Information gain itself is a strong predictor of task efficiency: a one-standard-deviation increase in Bayesian IG corresponds to a 43% reduction in expected game length, about twice the effect size of the ConceptNet-based IG (19%). By constraining LLaMA to avoid repeated question types or to ask only open-ended questions, we increase its success rate from 39.4% to 80.0% and from 39.4% to 97.4% respectively, greatly improving performance without architectural changes. Finally, when we apply our Bayesian IG metric post hoc to human-generated dialogues, we observe high correlations with game efficiency (Spearman $\rho = -0.95$ for experts and $\rho = -0.90$ for naive participants), exceeding correlations seen in model outputs. This suggests that the metric captures a domain-general notion of question informativeness, rather than merely reflecting model-specific uncertainty estimates. To support replication and future research, we release the **GuessingGame**, including code, prompts, and evaluation scripts.[1]

In summary, our contributions are:

1. We introduce GuessingGame, a novel open-domain, open-ended protocol for evaluating LLMs as strategic question-askers.
2. We propose two complementary information gain metrics: a Bayesian belief-tracking method using LLM-scored relevance over semantic concepts, and an entropy-based method grounded in ConceptNet.
3. We show that these metrics not only predict performance across humans and models, but also support interpretable diagnosis and prompt-level interventions that significantly improve model behavior.

2 Related Work

LLMs as Question Askers. Recent work explores LLMs as question-askers, often using the 20 Questions game (Walsorth, 1882) to assess strategic behavior. Gains are shown with belief tracking (Bertolazzi et al., 2023), reinforcement learning (Zhang et al., 2024), and preference tuning (Mazzaccara et al., 2024). Role-reversal (Noever and McKee, 2023) and ambiguity-resilient setups (Chen et al., 2024b) probe robustness, but remain domain-bounded or structured. Applied work in education (Chen et al., 2024a), healthcare (Li et al., 2024), and preference inference (Piriyakulkij et al., 2023) focuses on single-turn clarification under known contexts. Prompting strategies like Rephrase-and-Respond (Deng et al., 2024) and abstention-aware querying (Li et al., 2024) improve specificity but do not address long-horizon strategy. Most prior work is either closed-domain or non-strategic, whereas our evaluation protocol is open and strategic.

Information Gain and Strategic Reasoning. Effective questioning reduces uncertainty, and its utility is often quantified using expected information gain (EIG), entropy, or KL divergence. For example, Mazzaccara et al. (2024) leverage direct preference optimization (DPO) to fine-tune models that prefer more informative questions; Piriyakulkij et al. (2023) employ entropy-based acquisition functions to select questions that maximize uncertainty reduction about user preferences; and Hu et al. (2024) use forward-planning strategies that anticipate which queries will yield the most diagnostic responses. Symbolic reasoning approximations such as program sampling (Grand et al., 2024), belief filtering (Keh et al., 2023), and commonsense graph traversal (Zhao et al., 2023) further enable structured search over candidate spaces to generate or evaluate useful questions. These approaches typ-

[1] https://github.com/cincynlp/GuessingGame

ically operate in closed or well-structured domains. In contrast, we evaluate question quality without predefined answer spaces or acquisition objectives.

Reasoning About Objects in Language Models. Several studies probe whether LLMs encode object attributes, affordances, and physical reasoning. Benchmarks like NEWTON (Wang et al., 2023), PROST (Aroca-Ouellette et al., 2021), and TEXT2AFFORD (Adak et al., 2024) show that while models can reason abstractly, they often fail on concrete or uncommon affordances. In parallel, prior work has demonstrated that leveraging function knowledge supports object-use inference and visual activity recognition (Jiang and Riloff, 2022, 2023). Bertolazzi et al. (2023) finds that LLMs improve object identification when guided to reason over feature spaces. We extend this line of research by evaluating how models apply such knowledge in open-ended, multi-turn settings.

3 Methodology

We formalize the GuessingGame protocol and describe its implementation with LLMs. Our goal is to evaluate how effectively models gather information, not just whether they guess correctly. To support this, we introduce a multi-agent framework, evaluation metrics, and a question-type taxonomy for analysis and prompt-level control.

3.1 Task Formulation

GuessingGame is played by three agents: *Oracle*, *Guesser*, and *Checker*, instantiated as separate LLM instances to prevent information leakage. *Oracle:* Privy to a secret physical object drawn from an object corpus, the Oracle answers every question posed by the Guesser. *Guesser:* Asks questions about the Oracle's object to identify it. *Checker:* Classifies each Guesser query (by question type) and enforces any experiment-specific restrictions.

A single game proceeds in alternating turns. *Question Generation:* The Guesser asks a question based on the full dialogue history. *Validity Check:* The Checker verifies that the question adheres to all applicable constraints (e.g., "only attribute questions"). If a constraint is violated or if the Guesser attempts to ask "What is the object?" or a similarly trivializing question, it is prompted to revise the query (see Appendix D for validation details). *Oracle Response:* If the question is valid, the Oracle responds. If the Guesser makes a correct direct guess, the game ends; otherwise, play continues.

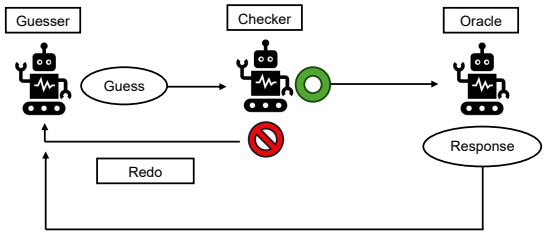

Figure 2: Overview of one GuessingGame round: Guesser asks, Checker validates, Oracle responds.

The game ends when the Guesser correctly names the object or after 50 turns (failure).

Formally, let $\mathcal{O}$ be the set of possible objects, $\mathcal{Q}$ the space of queries, and $\mathcal{A}$ the space of Oracle responses. A single game runs for up to $T_{\max}$ turns (we set $T_{\max} = 50$ for all our experiments). At turn t, the Guesser generates a question $Q_t = \text{Guesser}(H_{t-1})$, and the Oracle returns an answer $A_t = \text{Oracle}(Q_t, o^*)$, where $o^* \in \mathcal{O}$ is the secret object and $H_{t-1} = \{(Q_1, A_1), \ldots, (Q_{t-1}, A_{t-1})\}$ is the full dialogue history up to turn t. The game ends successfully at turn $T \leq T_{\max}$ if $Q_T =$ "`Is it a` $\hat{o}$?" and the Oracle answers "Correct." Otherwise it is a failure after $T_{\max}$.

Evaluation Metrics. We evaluate model performance using two primary metrics. First, *Success Rate (SR)* measures the proportion of games in which the Guesser successfully identifies the target object, reflecting overall task accuracy. Second, *Average Number of Questions (ANQ)* calculates the mean number of questions asked in successful games, indicating the model's efficiency. We define success rate as $\text{SR} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{1}(\text{game}_i \text{ succeeds})$ and average number of questions as $\text{ANQ} = \frac{1}{|\mathcal{S}|}\sum_{i \in \mathcal{S}} T_i$, where N is the total number of games, $\mathcal{S}$ is the set of successful games and T_i is the number of turns in game i.

3.2 Question Types

Rosch et al. (1976) showed that humans prefer "basic" category questions that maximize diagnostic features. Motivated by focused studies that target most common object-knowledge—*functions* (Chao et al., 2015; Jiang and Riloff, 2021), *locations* (Collell et al., 2018; Jiang and Riloff, 2018; Xu et al., 2018), *physical attributes* (Forbes and Choi, 2017; Tandon et al., 2017), and *category/taxonomic relations* (Suchanek et al., 2007; Shwartz et al., 2016)—we adopt these question types as the principal axes of inquiry:

Attribute questions gather physical features (shape, size, color). *Ex: What color is the object?*
Function questions probe purpose, unlocking causal or affordance-based reasoning. *Ex: Is the object used for communication?*
Location questions tap into contextual priors; knowing where something lives often reveals what it is. *Ex: Is the object typically found indoors?*
Category questions leverage taxonomic knowledge, asking "Is it a kind of X?" to traverse semantic hierarchies. *Ex: Is the object an instrument?*
Direct guesses commit to a hypothesis, serving as a binary test that can immediately terminate the search. *Ex: Is the object a table?*

Together, these question types capture key dimensions of inquiry: sensory grounding (Attribute), causal reasoning (Function), contextual inference (Location), conceptual classification (Category), and decisive hypothesis testing (Direct).

4 Measuring Information Gain

While success rate and question count reflect overall task performance, they do not capture how much each question reduces uncertainty. To address this, we introduce two complementary measures of *information gain* (IG) that evaluate the utility of individual question-answer pairs.

Information-theoretic approaches have long guided questioning strategies in 20 Questions-style tasks (Dagan et al., 2017; Bertolazzi et al., 2023), typically assuming a fixed candidate set. But these assumptions break down in open-domain settings like ours. Instead, we propose two methods to measure the information gain: (1) a *Bayesian belief-tracking* model that updates a distribution over semantic concepts using scores from an Interpreter LLM, and (2) a symbolic *entropy-based metric* that uses ConceptNet (Speer et al., 2017) to filter candidates based on answer-implied assertions.

4.1 Bayesian Belief Update

Intuitively, when playing the GuessingGame, a good answer should shift our "belief" about which objects remain plausible: a good question will eliminate unlikely candidates and boost the likelihood of those that fit the evidence. Since our GuessingGame is open-domain (object candidate list is not provided), it is not plausible to measure the probability distribution of each candidate during the game. Inspired by Smith et al. (2023), which prioritizes belief shifts over latent hypotheses, we measure a probability distribution over belief concepts (instead of potential objects) and update it whenever we observe a new answer. For example, if we know the hidden object is made of metal (concept), then it is unlikely to be clothing. By framing each answer as "soft evidence" for or against particular concepts, we can use a Bayesian-style update rule to track how uncertainty changes over time.

Interpreter LLM. To create a belief distribution, we introduce an *Interpreter* model—an LLM which is prompted to take the latest question and answer as input and returns a scored list of concepts, $\mathcal{S}_t = \{(c_i, r_i)\}_{i=1}^{m}$, where $r_i \in (-1, 1)$. Each concept c_i represents a physical or functional property (e.g., *metal*, *kitchen appliance*, *man-made*), and each score r_i indicates how strongly the answer supports or contradicts that concept. We treat negative scores as evidence against a concept and relabel them as negations, e.g., a score of -0.8 for *plastic* becomes "*not plastic*" with score 0.8. This framework follows the intuition behind verbalized confidence scoring (Yang et al., 2024), to assign explicit relevance scores to candidate concepts.

Belief Update. To achieve an *open-world* setting we do not assume a predefined concept pool. We begin each game with an *empty belief state* $b_0(c)$—no concept receives any probability mass until it is first introduced by the Interpreter. Evidence accrued during the dialogue then builds the posterior from scratch using a log-linear update:

$$\tilde{b}_{t+1}(c) = \begin{cases} b_t(c) \cdot \exp(\alpha \cdot r_c), & \text{if } c \in b_t \\ \exp(\alpha \cdot r_c), & \text{otherwise} \end{cases}$$
$$b_{t+1}(c) = \frac{\max(\tilde{b}_{t+1}(c), \varepsilon)}{\sum_{c'} \max(\tilde{b}_{t+1}(c'), \varepsilon)} \quad (1)$$

Here, $\alpha > 0$ controls the influence of the evidence (we use $\alpha = 1$; see Appendix F), and $\varepsilon = 10^{-12}$ prevents zero mass. A pruning threshold is used to discard concepts whose posterior mass falls below a fixed cutoff, preventing the belief state from being diluted by dozens of near-zero hypotheses and keeping the Guesser focused on the most plausible candidates. This formulation corresponds to a soft-evidence update consistent with *Jeffrey conditioning* (Jeffrey, 1965), treating each relevance score as a log-likelihood proxy. The exponential form is well-suited to our setting, where observations are uncertain, continuous-valued (i.e., LLM-scored), and no hard posterior is known. It

provides a smooth, monotonic shift toward concepts most consistent with the answer.

To measure how much the belief changed from one turn to the next, we compute the KL divergence between the updated belief and the prior:

$$\begin{aligned} \mathrm{IG}_t &= D_{\mathrm{KL}}(b_{t+1} \parallel b_t) \\ &= \sum_c b_{t+1}(c) \log \frac{b_{t+1}(c)}{\max(b_t(c), \varepsilon)}. \end{aligned} \tag{2}$$

This value increases when the distribution becomes more focused, i.e., the model becomes more confident in a smaller set of hypotheses. This reflects the principle that informativeness arises when answers induce meaningful belief shifts over predictions, consistent with work on prediction-oriented acquisition functions (Smith et al., 2023).

Example. At turn t, the Guesser asks "What material is it made of?" and the Oracle replies "It's shiny and metallic." The Interpreter processes this exchange and outputs relevance scores for high-level concepts: `metal`: 0.9, `steel`: 0.7, `aluminum`: 0.6. These scores are treated as soft evidence in the belief update, boosting concepts that align with the answer using the log-linear update. Concepts not mentioned (e.g., `plastic`, `wood`) retain their scores and become down-weighted during normalization. This shifts the belief distribution toward more plausible hypotheses and yields a gain in information measurable by KL divergence.

4.2 Entropy-Based Information Gain

Alternative to our Bayesian belief-tracking approach, we propose a method for estimating information gain based on uncertainty reduction in an existing knowledge graph. If the Oracle's answer implies that the object likely has a certain property (e.g., *sharp*), we can prune candidates that lack that property for measurable entropy reduction.

We use ConceptNet (Speer et al., 2017), a large commonsense knowledge graph where nodes are natural language concepts and directed edges encode semantic relations such as `IsA`, `MadeOf`, `UsedFor`, and `HasProperty`. For example, (`/r/HasProperty`, `/c/en/knife`, `/c/en/sharp`) and (`/r/UsedFor`, `/c/en/knife`, `/c/en/cutting`). This lets us ground free-form Oracle answers in a symbolic space of semantic hypotheses.

Matching Answers to Assertions. Given an Oracle response A_t, we convert it into an embedding vector $\mathbf{v}_A$ using a pre-trained model all-MiniLM-L6-v2 from the Sentence Transformers library (Reimers and Gurevych, 2019). Each ConceptNet concept label is also embedded into a vector $\mathbf{v}_c$. We then compute the cosine similarity between the Oracle response and each concept as $\mathrm{sim}(A_t, c) = \frac{\mathbf{v}_A \cdot \mathbf{v}_c}{\|\mathbf{v}_A\| \, \|\mathbf{v}_c\|}$. This allows us to identify the concepts most semantically related to the Oracle's answer. For all concepts c where $\mathrm{sim}(A_t, c) \geq \tau$ (we use $\tau = 0.60$; see Appendix E), we collect all ConceptNet edges that end in c: that is, we extract assertions of the form (r, o, c), where r is a relation and o is a possible object. This gives us a set of assertions (r, c) that are semantically implied by the answer.

At each turn t, we maintain a set $\mathcal{D}_t$ of remaining candidate objects. Initially this is all possible objects in ConceptNet. After each Oracle response, we shrink this set based on the matched assertions. For each assertion (r, c), we retrieve the subset of objects consistent with that assertion: $\mathcal{Y}_t^{(r,c)} = \{o \in \mathcal{D}_t \mid (r, o, c) \in \text{ConceptNet}\}$. We then define the updated candidate set as $\mathcal{D}_{t+1} = \bigcup_{(r,c)} \mathcal{Y}_t^{(r,c)}$, the union of all "yes-sets". In other words, we retain any object $o \in \mathcal{D}_t$ that matches *at least one* of the answer-implied assertions; objects that match none are filtered out.

Measuring Entropy Reduction. We assume a uniform prior over the current candidate set $\mathcal{D}_t$, so the initial uncertainty is $H_{\text{prior}} = \log_2 |\mathcal{D}_t|$. After applying the filter, the new candidate set is $\mathcal{D}_{t+1}$, and the updated uncertainty becomes $H_{\text{post}} = \log_2 |\mathcal{D}_{t+1}|$. We define information gain as the drop in entropy:

$$\mathrm{IG}_t = H_{\text{prior}} - H_{\text{post}} = \log_2 \frac{|\mathcal{D}_t|}{|\mathcal{D}_{t+1}|}, \tag{3}$$

which reflects how much the question-answer exchange reduced the size, and thus uncertainty, of the hypothesis space.

This entropy-based metric captures how ConceptNet knowledge prunes unlikely candidates for the secret object. The candidate pool shrinks each turn, since $\mathcal{D}_{t+1} \subseteq \mathcal{D}_t$, guaranteeing non-negative information gain. In Section 5.2, we compare this method to our Bayesian KL metric and show that it correlates with convergence, albeit less strongly.

Design Tradeoffs. While both information gain metrics estimate how much a question reduces uncertainty, they differ in assumptions, scalability, and cost. The Bayesian method is much more flexible, requiring no fixed knowledge base, and handles implicit properties and unstructured domains.

Condition	SR (%)	ANQ
Closed-Ended	32.1 ± 3.12	25.0 ± 1.35
Open-Ended	**39.4 ± 3.26**	**23.3 ± 1.36**

Table 1: LLaMA-3.3 70B performance under open-ended vs. binary-only questions. 95% confidence intervals shown. SR–success rate, ANQ–average number of questions.

Type	Ratio (%)	Bayes IG (σ)	Entropy IG (σ)
Attribute	37.6	**+0.19**	**+0.24**
Direct	21.7	+0.08	-0.13
Category	14.0	-0.01	-0.00
Function	23.6	-0.07	-0.18
Location	2.90	-0.19	+0.08
Open-Ended	5.90	+0.12	**+0.03**
Closed-Ended	94.1	-0.12	-0.03

Table 2: Proportion and mean IG per question type and format, reported as standard deviations from the overall mean IG.

However, it is computationally expensive and depends on the calibration of the Interpreter model. In contrast, the ConceptNet-based method is more efficient and model-free, relying on sentence embeddings and graph lookups to prune the candidate set. But it is limited by ConceptNet's coverage and may miss properties not explicitly encoded.

5 Results

We evaluate our GuessingGame protocol across various settings, each run for a total of 858 games. Unless otherwise noted, all agents—the Guesser, Oracle, and Checker—were instantiated with LLaMA-3.3 70B and a temperature of 0.6.

Object Corpus. We draw our secret objects from Jiang and Riloff (2021), a broad collection of everyday objects annotated with their typical functions. To obtain a clean set of standalone objects, we exclude high-level categories (e.g., *apparel*, *appliance*) that do not denote specific objects, and de-duplicate synonymous entries (e.g., *axe* vs. *ax*). The resulting corpus consists of 858 distinct objects which we test in all experiments.

5.1 GuessingGame Results

We begin our evaluation by assessing LLM performance on the core GuessingGame task: identifying a hidden object through multi-turn, free-form dialogue. Table 1 summarizes baseline results for LLaMA-3.3 70B under two conditions: the standard, unconstrained setting in which the

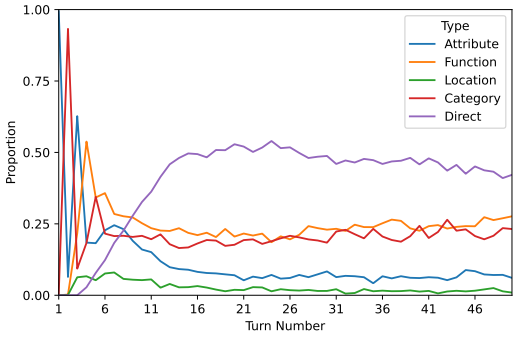

Figure 3: Distribution of question types by turn. Later turns reflect fewer games, as many conclude early, so proportions in later rounds are based on smaller samples.

Guesser may ask any type of question, and a more traditional closed-ended variant restricted to yes/no queries. In the open-ended setting, the model achieves a 39.4% success rate. When constrained to binary prompts, performance drops to 32.1%.

Intuitively, this makes sense: open-ended prompts elicit more complete answers, while binary questions may convey no new information depending on the response. For example, "What material is the object made of?" yields a useful answer regardless of the object, whereas "Is it metal?" is only informative if the answer is "yes."

Our information gain metrics reinforce this interpretation. We measure the average IG per question type and report them as standard deviations from the overall mean IG. As shown in Table 2, open-ended questions yield substantially higher average IG than closed-ended ones ($+0.12\sigma$ vs. -0.12σ under the Bayesian metric).

Despite their clear advantage, open-ended questions are rarely used: only 5.9% of all questions were open-ended. This suggests a missed opportunity and motivates our later experiments, which test whether prompting strategies can encourage more informative, high-yield questions.

Question-Type. Effective inquiry often depends on the type of question being asked. Cognitive psychology research has shown that concrete, perceptual questions (e.g., about size or material) tend to be more diagnostic than abstract or contextual questions (Rosch et al., 1976).

Figure 3 shows how our question types (see Section 3.2) are distributed over the course of the gameplay of the standard GuessingGame task. Early turns are dominated by exploratory questions, especially Attribute and Function, while later rounds shift toward Direct guesses. This reflects a shift

Question Type	SR (%)	ANQ
All Types	**39.4 ± 3.26**	**23.3 ± 1.36**
Attribute-Only	35.8 ± 3.20	23.6 ± 1.30
Function-Only	31.0 ± 3.09	24.3 ± 1.28
Location-Only	18.4 ± 2.59	24.5 ± 9.40

Table 3: LLaMA-3.3 70B GuessingGame performance when limited to specific question types.

from exploration to hypothesis testing.

To isolate the utility of each question type, we run a controlled experiment where the Guesser is restricted to asking only one type of information-seeking question: *Attribute*, *Function*, or *Location*, while still permitting *Direct* guesses. Table 3 shows that Attribute-only questions yield a 35.8% success rate, nearly matching the full-question baseline (39.4%). Function-only and Location-only conditions perform worse (31.0% and 18.4%, respectively). This disparity likely reflects differences in expressive range: most objects afford only one or two meaningful function or location queries (e.g., "What is it used for?", "Where is it found?"), whereas Attribute questions have many aspects to probe (e.g., size, shape, material, and color).

These behavioral trends are reflected in our information gain metrics. As shown in Table 2, Attribute questions achieve the highest average information gain (+0.19σ Bayesian, +0.24σ entropy), while Function and Location questions perform worse. These findings directly align with the Bayesian IG rankings (Attribute>Function>Location), suggesting that the Bayesian metric captures the per-type informativeness of questions with high fidelity.

5.2 Information Gain Comparison

We compare our two information gain (IG) metrics, Bayesian belief updates and ConceptNet-based entropy reduction, by asking: does higher IG predict faster convergence to the correct object?

Spearman Correlation. To evaluate if IG predicts success, we compute Spearman correlation ρ between mean IG per round and total game length. Bayesian IG shows a stronger correlation in Spearman correlation with total number of questions as opposed to Entropy-based IG: $\rho = -0.63$ ($p = 1.51 \times 10^{-13}$) vs. $\rho = -0.21$ ($p = 2.73 \times 10^{-21}$). This suggests Bayesian IG better reflects long-term informativeness trends.

Accelerated Failure Time Model. To capture the turn-level predictive power of IG, we apply an *Accelerated Failure Time* (AFT) model, commonly used in survival analysis. AFT models estimate how covariates directly scale expected time-to-event, in this case, the number of turns until the Guesser succeeds. AFT operates in log-time, expressing the logarithm of expected duration as a linear function of predictors. Coefficients can be exponentiated to interpret the multiplicative effect of each unit increase in a predictor.

IG Metric	AFT Coefficient (p)	Spearman ρ
Bayesian	-0.57	-0.63
Entropy	-0.21	-0.25

Table 4: Comparison of IG metrics. AFT coefficients reflect the log-linear effect of IG on game length; Spearman correlations are computed between average IG and number of rounds to completion. Negative values indicate that higher IG predicts faster convergence. All coefficients are significant at the $p < 0.001$ level.

In our analysis, Bayesian IG yields a strong negative effect ($\beta = -0.57$, $p = 1.77 \times 10^{-7}$), meaning that for every one standard deviation increase in IG, the expected number of turns is scaled by a factor of $e^{-0.57} \approx 0.57$, a **43% reduction**. Entropy-based IG also has a significant effect ($\beta = -0.21$, $p = 1.25 \times 10^{-12}$), corresponding to a **19% reduction** in expected game length ($e^{-0.21} \approx 0.81$).

Both metrics significantly predict task convergence, but Bayesian IG consistently outperforms entropy-based IG in both correlation and effect size. While entropy-based IG provides a fast, model-free signal grounded in commonsense pruning, Bayesian IG offers a more descriptive and flexible measure of question utility.

6 Analysis

We analyze two complementary aspects of performance on the GuessingGame protocol: (1) how simple prompting interventions affect model behavior, and (2) how different LLMs compare in terms of strategic questioning ability. In both cases, Bayesian information gain serves as a useful measure for interpreting and explaining observed differences in performance.

Improving Behavior through Prompting Constraints. Qualitatively, we noticed a common failure mode in GuessingGame which we call *enumerative questioning*, where the Guesser issues a sequence of near-identical queries that vary only slightly in content (e.g., "Is it made in Ohio?",

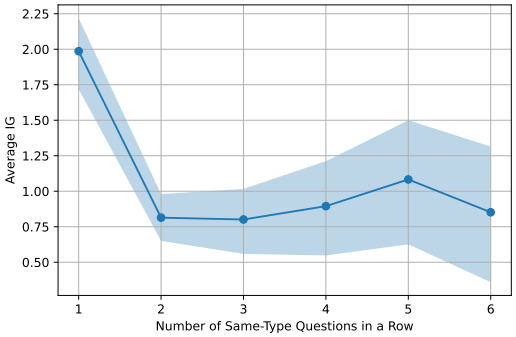

Figure 4: Average Bayesian information gain by number of consecutive same-type questions. 95% CI shown.

Model / Condition	SR	ANQ	Spearman ρ
LLaMA-3.3 70B			
Standard	39.4 ± 3.30	23.3 ± 1.30	-0.63
Repeat Constraint	80.0 ± 2.70	12.6 ± 1.00	-0.69
Forced Open	97.4 ± 1.06	8.30 ± 0.56	-0.51
GPT-4o			
Standard	64.1 ± 3.20	20.1 ± 1.17	-0.33
Forced Open	98.5 ± 0.83	6.90 ± 0.48	-0.45
Gemini 2.0 Flash-Lite			
Standard	74.1 ± 2.92	16.5 ± 1.28	-0.43
Forced Open	87.3 ± 2.22	13.4 ± 1.09	-0.47

Table 5: LLM performance under different prompting strategies and models. Spearman coefficients (ρ) show correlation between the average IG per question per game and the length of the game, with significance at the $p < 0.001$ level.

Group	SR (%)	ANQ	Spearman ρ
Experts	96.3 ± 2.50	7.00 ± 1.00	-0.95
Naive	88.8 ± 6.90	9.24 ± 2.20	-0.90

Table 6: Human performance on the GuessingGame task. 95% confidence intervals shown. Spearman coefficients represent the correlation between the average Information Gain per question per game and length of the game, with significance at the $p < 0.001$ level.

"...in New York?", "...in Germany?"). To assess the impact of this behavior, we analyzed how information gain changes under these conditions. As shown in Figure 4, average IG drops sharply when models repeat the same type across consecutive turns, indicating diminishing returns. To address this, we introduced a repeat-type prompting constraint that prevents back-to-back questions of the same type. This simple intervention leads to substantial improvement: LLaMA-3.3 70B's success rate more than doubles, rising from 39.4% to 80.0%.

Our second intervention targets question format. As shown in Table 2, open-ended questions yield higher IG than binary (yes/no) questions. Yet, despite their higher informativeness, open-ended questions were rarely used in LLaMA's default behavior. So, we introduced a forced open-endedness prompt constraint, limiting the Guesser to free-form questions except for final direct guesses—increasing the success rate to 97.4%. Together, these findings demonstrate that targeted prompting strategies, motivated by IG trends, can substantially improve question quality and task performance.

Model Comparisons. We evaluate several LLMs on GuessingGame to assess their ability to ask informative, goal-directed questions.

Table 5 shows that proprietary models such as GPT-4o (OpenAI et al., 2024b) and Gemini 2.0 Flash-Lite (Team et al., 2025) perform better out of the box, with them achieving a 64.1% and 74.1% success rate respectively. In contrast, LLaMA-3.3 70B, under default prompting, has a 39.4% success rate. When prompted to adopt an open-ended strategy via constraints enforcing question diversity and free-form formats, every model shows a striking improvement in performance. These results underscore a key distinction between *capability* and *behavior*. Weaker models like LLaMA can match, or even outperform, stronger models, *if* prompted to ask better questions.

However, regardless of models and settings, higher per-turn information gain is consistently associated with shorter games and greater accuracy.

Human Performance Comparison. To contextualize LLM performance, we conducted a small-scale evaluation with human participants. Two of the paper's authors (familiar with the task design) and two unpaid naive volunteers (with no prior exposure) each completed 40 games under the standard 50-turn limit, yielding a total of 160 games. Results are shown in Table 6.

We applied our Bayesian information gain metric post hoc to the human-generated dialogues. Per-turn IG was a very strong predictor of game efficiency, with Spearman $\rho = -0.95$ for experts and $\rho = -0.90$ for naive participants, exceeding the corresponding values observed in LLMs.

These results raise a key question about what the metric is actually measuring. While our IG formulation is motivated by verbalized relevance scores intended to approximate belief updates from LLMs (Yang et al., 2024), its consistently high correlation with both human and model performance suggests

it may not reflect internal uncertainty. Rather, it *appears* here to function as a general-purpose measure of question informativeness.

7 Conclusion

We present GuessingGame, a protocol for evaluating large language models as question-askers in open-ended, open-domain settings. Framed as an interactive guessing task, it enables principled assessment of question quality using Bayesian belief updates and entropy-based metrics. Our results show that question-asking is both measurable and improvable. This work lays the foundation for richer evaluations of curiosity, exploration, and strategy in language models.

Limitations

While GuessingGame provides a novel and rigorous framework for evaluating question-asking behavior in large language models (LLMs), several limitations merit discussion.

External vs. Internal Belief Modeling. Our Bayesian information gain metric is computed via an external belief-tracking mechanism rather than derived from internal model states. While this allows for interpretability and post hoc analysis across any model's output or question, it does not reveal whether LLMs internally represent beliefs or update them coherently across turns. Our metric measures observable behavior, not necessarily latent cognition. Future work should investigate whether these externally modeled belief updates align with a model's internal representations, potentially leveraging adequacy criteria proposed by Herrmann and Levinstein (2024).

Dependence on the Interpreter LLM. Our Bayesian IG metric depends on the accuracy and calibration of an *Interpreter LLM*, which scores relevance of answer-implied concepts. This introduces a second-order model dependency that may inject bias or noise. If the Interpreter misjudges the semantic content of an answer, belief updates may be misleading. While we mitigate this through normalization and smoothing, future work should validate alternative interpreters, explore ensemble methods, or benchmark against human-labeled relevance scores.

Interpretive Status of Bayesian IG. Our results suggest that Bayesian information gain is a strong predictor of task efficiency across both LLMs and humans. In particular, its post hoc application to human-generated dialogues yields striking correlation with game performance. However, we do not claim to have formally established that this metric constitutes a domain-general or human-aligned measure of question informativeness. While the observed correlations are promising, they do *not* prove that the metric captures the same cognitive principles humans use when formulating questions, nor that it generalizes beyond the GuessingGame context. Further work is needed to test whether this metric aligns with human judgments of informativeness across diverse tasks, question formats, and domains. Our current findings should thus be interpreted as *preliminary* evidence that Bayesian IG *could* serve as a general-purpose metric, not a definitive validation.

Knowledge Base Coverage for Entropy-Based IG. Our entropy-based metric depends on ConceptNet's graph structure to filter candidate objects. However, ConceptNet has limited coverage for niche or multi-functional objects and contains sparse or noisy edges for some object-property pairs. This makes the metric more reliable for common objects but potentially brittle in low-resource or specialized domains. Additionally, the reliance on static embeddings for semantic matching may overlook subtle answer nuances not captured by cosine similarity.

Limited Domain Scope. While our experiments focus exclusively on everyday, prototypical objects, the GuessingGame framework is general and could be instantiated over a wide range of object sets. For example, the task could be adapted to diagnostic domains by using diseases as the hidden concepts and simulating symptom-based queries. Similarly, it could be applied to scientific discovery, legal reasoning, or strategic gameplay where the hidden target represents a theory, precedent, or opponent strategy. In this work, we restrict our scope to concrete, physical artifacts to ensure interpretability and controlled analysis, but future work could explore more abstract or high-stakes domains.

Acknowledgments

This work benefited greatly from the discussions of the CincyNLP group. We further thank the anonymous EMNLP reviewers for their careful reading and thoughtful feedback. We also thank volunteer game players, Jim and Marylee Vennemeyer.

References

Sayantan Adak, Daivik Agrawal, Animesh Mukherjee, and Somak Aditya. 2024. Text2afford: Probing object affordance prediction abilities of language models solely from text. In *Proceedings of the 28th Conference on Computational Natural Language Learning (CoNLL 2024)*.

Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, , and W. Bruce Croft. 2019. Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2019)*.

Stéphane Aroca-Ouellette, Cory Paik, Alessandro Roncone, and Katharina Kann. 2021. Prost: Physical reasoning of objects through space and time. In *Findings of the Association for Computational Linguistics (ACL-IJCNLP 2021)*.

Leonardo Bertolazzi, Davide Mazzaccara, Filippo Merlo, and Raffaella Bernardi. 2023. Chatgpt's information seeking strategy: Insights from the 20-questions game. In *Proceedings of the 16th International Natural Language Generation Conference (INLG 2023)*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, and 13 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33 (NeurIPS 2020)*.

Yu-Wei Chao, Zhan Wang, Rada Mihalcea, and Jia Deng. 2015. Mining semantic affordances of visual object categories. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2015)*.

Yuyan Chen, Chenwei Wu, Songzhou Yan, Panjun Liu, and Yanghua Xiao. 2024a. Dr.Academy: A benchmark for evaluating questioning capability in education for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*.

Yuyan Chen, Tianhao Yu, Yueze Li, Songzhou Yan, Sijia Liu, Jiaqing Liang, and Yanghua Xiao. 2024b. Do large language models have problem-solving capability under incomplete information scenarios? In *Findings of the Association for Computational Linguistics (ACL 2024)*.

Guillem Collell, Luc Van Gool, and Marie-Francine Moens. 2018. Acquiring common sense spatial knowledge through implicit spatial templates. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence (AAAI 2018)*.

Yuval Dagan, Yuval Filmus, Ariel Gabizon, and Shay Moran. 2017. Twenty (simple) questions. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing (STOC 2017)*.

Yihe Deng, Weitong Zhang, Zixiang Chen, and Quanquan Gu. 2024. Rephrase and respond: Let large language models ask better questions for themselves. *Preprint*, arXiv:2311.04205.

Maxwell Forbes and Yejin Choi. 2017. Verb physics: Relative physical knowledge of actions and objects. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL 2017)*.

Gabriel Grand, Valerio Pepe, Jacob Andreas, and Joshua B. Tenenbaum. 2024. Loose lips sink ships: Asking questions in battleship with language-informed program sampling. In *Proceedings of the Annual Meeting of the Cognitive Science Society (CogSci 2024)*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Daniel A. Herrmann and Benjamin A. Levinstein. 2024. Standards for belief representations in LLMs. *Minds and Machines*, 35(1).

Zhiyuan Hu, Chumin Liu, Xidong Feng, Yilun Zhao, See-Kiong Ng, Anh Tuan Luu, Junxian He, Pang Wei Koh, and Bryan Hooi. 2024. Uncertainty of thoughts: Uncertainty-aware planning enhances information seeking in large language models. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*.

Richard C. Jeffrey. 1965. *The Logic of Decision*. University of Chicago Press, New York, NY, USA.

Tianyu Jiang and Ellen Riloff. 2018. Learning prototypical goal activities for locations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*.

Tianyu Jiang and Ellen Riloff. 2021. Learning prototypical functions for physical artifacts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*.

Tianyu Jiang and Ellen Riloff. 2022. Identifying physical object use in sentences. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP 2022)*.

Tianyu Jiang and Ellen Riloff. 2023. Exploiting commonsense knowledge about objects for visual activity

recognition. In *Findings of the Association for Computational Linguistics: ACL 2023 (Findings of ACL 2023)*.

Sedrick Keh, Justin T. Chiu, and Daniel Fried. 2023. Asking more informative questions for grounded retrieval. *Preprint*, arXiv:2311.08584.

Shuyue Stella Li, Vidhisha Balachandran, Shangbin Feng, Jonathan S. Ilgen, Emma Pierson, Pang Wei Koh, , and Yulia Tsvetkov. 2024. Mediq: Question-asking LLMs and a benchmark for reliable interactive clinical reasoning. In *Advances in Neural Information Processing Systems 37 (NeurIPS 2024)*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *Preprint*, arXiv:1907.11692.

Davide Mazzaccara, Alberto Testoni, and Raffaella Bernardi. 2024. Learning to ask informative questions: Enhancing LLMs with preference optimization and expected information gain. In *Findings of the Association for Computational Linguistics: EMNLP 2024 (Findings of EMNLP 2024)*.

David Noever and Forrest McKee. 2023. Chatbots as problem solvers: Playing twenty questions with role reversals. *Preprint*, arXiv:2301.01743.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, and 262 others. 2024a. Gpt-4 technical report. *Preprint*, arXiv:2303.08774.

OpenAI, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, Alex Nichol, and 400 others. 2024b. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Top Piriyakulkij, Volodymyr Kuleshov, and Kevin Ellis. 2023. Asking clarifying questions using language models and probabilistic reasoning. In *Foundation Models for Decision Making Workshop (NeurIPS 2023)*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing (EMNLP 2019)*.

Eleanor Rosch, Carolyn B Mervis, Wayne D Gray, David M Johnson, and Pennt Boyes-Braem. 1976. Basic objects in natural categories. *Cognitive psychology*, 8(3).

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*.

Freddie Bickford Smith, Andreas Kirsch, Sebastian Farquhar, Yarin Gal, Adam Foster, and Tom Rainforth. 2023. Prediction-oriented bayesian active learning. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics (AISTATS 2023)*.

Robert Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI 2017)*.

Fabian M. Suchanek, Gjergji Kasneci, and Gerhard Weikum. 2007. Yago: A core of semantic knowledge unifying wordnet and wikipedia. In *Proceedings of the 16th International Conference on World Wide Web (WWW 2007)*.

Niket Tandon, Gerard de Melo, and Gerhard Weikum. 2017. WebChild 2.0 : Fine-grained commonsense knowledge distillation. In *Proceedings of ACL 2017, System Demonstrations*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy Lillicrap, Angeliki Lazaridou, and 1332 others. 2025. Gemini: A family of highly capable multimodal models. *Preprint*, arXiv:2312.11805.

M.T. Walsorth. 1882. *Twenty Questions: A Short Treatise on the Game to which are Added a Code of Rules and Specimen Games for the Use of Beginners*. Holt.

Lei Wang, Chen Ma, Xueyang Feng, Zeyu Zhang, Hao Yang, Jingsen Zhang, Zhiyuan Chen, Jiakai Tang, Xu Chen, Yankai Lin, Wayne Xin Zhao, Zhewei Wei, and Jirong Wen. 2024. A survey on large language model based autonomous agents. *Frontiers of Computer Science*, 18(6).

Yi Ru Wang, Jiafei Duan, Dieter Fox, and Siddhartha Srinivasa. 2023. Newton: Are large language models capable of physical reasoning? In *Findings of the Association for Computational Linguistics (EMNLP 2023)*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, and 3 others. 2020. Huggingface's transformers: State-of-the-art natural language processing. *Preprint*, arXiv:1910.03771.

Frank F. Xu, Bill Yuchen Lin, and Kenny Zhu. 2018. Automatic extraction of commonsense LocatedNear knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*.

Daniel Yang, Yao-Hung Hubert Tsai, and Makoto Yamada. 2024. On verbalized confidence scores for LLMs. *Preprint*, arXiv:2412.14737.

Yizhe Zhang, Jiarui Lu, and Navdeep Jaitly. 2024. Probing the multi-turn planning capabilities of LLMs via 20 question games. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL 2024)*.

Zirui Zhao, Wee Sun Lee, and David Hsu. 2023. Large language models as commonsense knowledge for large-scale task planning. In *Advances in Neural Information Processing Systems 36 (NeurIPS 2023)*.

A Error Analysis

There are many different types of errors that can occur in a guessing game. Errors that we encountered are listed and described in this section.

- **Enumeration** is when the Guesser keeps asking very specific and similar questions, thus yielding minimal information. This is the most common error and often does not end naturally once it starts. It can be mitigated by encouraging the Guesser to ask high-information questions or by preventing repeated question types.
- The Oracle can give **Incorrect Responses**, usually due to misunderstanding the Guesser's question or misjudging its relevance to the object. Depending on the importance of the question, this can either have little impact or completely derail the game. There is no simple solution, as this error reflects the Oracle's incomplete or inconsistent object knowledge.
- A **Misleading Response** occurs when the Oracle gives a technically correct but easily misinterpreted answer. For example, if you can just barely hold an object in your hands, and the Oracle replies "yes" to "Can it be held in your hands?", the Guesser may incorrectly assume the object is much smaller. These subtle misunderstandings can misdirect the Guesser's strategy and reduce efficiency.
- **Hierarchy Mismatch** occurs when the Guesser fixates on the wrong level of semantic abstraction, either too specific or too general, relative to the Oracle's object. In some cases, the Guesser gets stuck distinguishing between fine-grained subtypes (e.g., "thermos," "canteen," "water bottle") when the correct answer is simply "container." In other cases, the Guesser asks questions that are too vague or high-level (e.g., "Is it an object?" or "Is it man-made?"), which fail to narrow the hypothesis space meaningfully. This mismatch often leads to inefficient questioning and can be difficult to recover from without stronger concept-level reasoning or hierarchical search strategies.

B Enumeration Analysis

Enumeration is the most common error and the one most likely to lead to failure. For each experiment,

Question Type	Enumeration Percent (%)
All Types	14.4
All Types k=1	**1.30**
All Types k=2	3.40
All Types Forced Open	3.60
Attribute Only	23.7
Function Only	44.4
Location Only	48.0

Table 7: Average percent of questions that are enumerations across different questions types using LLaMA-3.3 70B.

Model	Enumeration Percent (%)
GPT-4o	11.3
GPT-4o k=1	0.60
GPT-4o Forced Open	0.10
GPT-4o Forced Open k=1	**0.00**
Gemini	16.5
Gemini k=1	1.40
Gemini Forced Open	6.10
Gemini Forced Open k=1	**0.00**

Table 8: Average percent of questions that are enumerations for GPT-4o and Gemini 2.0 Flash-Lite.

we count the percentage of the Guesser's queries that were enumerations. We show these result in Table 7 and Table 8. The two type-restrictions naturally have the lowest enumeration rate, since enumeration is defined by repeatedly asking similar questions and type-restrictions prevent this. Experiments where the Guesser is restricted to asking one type of question increase enumeration by definition. The rates for function-only and location-only are significantly higher than the rest, while attributes-only is relatively low considering the constraint. This occurs because there are fewer ways to ask about an object's purpose or location than there are to ask about all of its attributes.

C Forced Open Questions

Since each model has a low inclination towards choosing open-ended questions when not prompted to (Table 9), we forced them to ask only open-ended questions with the exception of direct questions (which are necessary to complete the game). The results are shown in Table 10 and Table 11. There is a significant improvement across all experiments. An open-ended question is guaranteed to learn a new piece of information from every question, unlike closed-ended questions. Many objects can be identified by a few key aspects, such as primary function and location. Through only open-ended questions, these aspects can be learned in a few questions, allowing a quick victory, as demonstrated in this experiment. When these early open questions do not identify the object, this leads to an increased number of direct guesses. This means that the models' inherent reasoning capabilities affect the percent of open-ended questions.

Model	Open-Ended (%)
LLaMA	5.90
LLaMA k=1	41.8
LLaMA Forced Open	70.5
LLaMA Forced Open k=1	**77.3**
GPT	0.70
GPT k=1	11.9
GPT Forced Open	**73.7**
GPT Forced Open k=1	72.1
Gemini	7.50
Gemini k=1	32.9
Gemini Forced Open	24.4
Gemini Forced Open k=1	**59.8**

Table 9: Proportion of open-ended vs. closed-ended questions used by each model during standard GuessingGame gameplay. Open-ended questions elicit richer Oracle responses and are associated with higher information gain. LLaMA is LLaMA-3.3 70B, GPT is ChatGPT-4o, Gemini is Gemini 2.0 Flash-Lite.

Question Type	SR (%)	ANQ
All Types	97.4 ± 1.06	8.30 ± 0.56
All Types, k=1	98.1 ± 0.92	7.70 ± 0.48
All Types, k=2	$\mathbf{99.2 \pm 0.62}$	$\mathbf{8.40 \pm 0.62}$
Attribute-Only	70.2 ± 3.05	16.2 ± 1.11
Function-Only	63.6 ± 3.21	15.3 ± 1.13
Location-Only	53.5 ± 3.33	16.6 ± 1.25

Table 10: Performance of LLaMA-3.3 70B on the GuessingGame task with the forced-open constraint, where the Guesser is restricted to asking only open-ended questions (except for final direct guesses). **SR** denotes success rate, and **ANQ** is the average number of questions asked. Rows labeled *All Types* allow the Guesser to use any type of open-ended question. The k parameter denotes a **repeat-type constraint**, which limits the number of consecutive questions of the same type: *k=1* prohibits back-to-back questions of the same type, while *k=2* allows up to two in a row. Lower values of k enforce greater question-type diversity. Restricted rows (*Attribute-Only*, *Function-Only*, *Location-Only*) constrain the Guesser to a single type of open-ended question, revealing the relative informativeness of each question type when used in isolation.

Question Type	SR (%)	ANQ
GPT-4o	**98.5 ± 0.83**	**6.60 ± 0.42**
GPT-4o k=1	97.8 ± 0.99	6.90 ± 0.48
Gemini	87.3 ± 2.22	13.4 ± 1.09
Gemini k=1	97.7 ± 1.02	8.90 ± 0.75

Table 11: Performance of GPT-4o and Gemini 2.0 Flash-Lite with the forced-open constraint.

Approach	Acc.	P_{macro}	R_{macro}	$F1_{macro}$
Rule-Based Baseline	0.82	0.89	0.64	0.67
RoBERTa Classifier	0.96	0.96	0.87	0.90
LLM Checker	0.95	0.93	0.96	0.94

Table 12: Performance of different question-type checkers. Macro- and weighted-average precision (P), recall (R), and F1, plus overall accuracy.

The general difference between the types of questions is similar to previous experiments, though the gap between all types and restricted types is larger and the gap between the restricted types is smaller. This shows the benefit of diverse questions, as there is a limit to the amount of information to be gained from only one type of question.

D Checker Validation

Since we use our Checker LLM to enforce our experiment's parameters (e.g., function questions only), we experimentally validate that our Checker can correctly classify all question types (attribute, function, location, category). We manually annotated 1,000 questions randomly sampled from actual Guesser outputs from our experiment. There was an inter-annotator agreement of 0.88. We compare three approaches on human-annotated data: a rule-based baseline, a fine-tuned RoBERTa classifier (80/20 train/test split), and the prompt-based LLM Checker. Table 12 summarizes their macro-average performance and overall accuracy.

The rule-based system achieves an accuracy of 0.82 and suffers from low recall on less frequent types. A fine-tuned RoBERTa (Liu et al., 2019) classifier yields high overall accuracy at 0.96 and strong macro-F1 (0.90), demonstrating the task's learnability from moderate data. Our LLM Checker matches this performance (accuracy 0.95, macro-F1 0.94) without any additional fine-tuning, confirming that prompt-based classification is a reliable and maintenance-free choice for enforcing question-type constraints in GuessingGame.

RoBERTa Classifier Setup. We fine-tuned a `roberta-large` model using HuggingFace Transformers (Wolf et al., 2020) on an 80/20 stratified split of the 1,000 labeled examples. The model was trained for 10 epochs with a batch size of 8 using the AdamW optimizer and the default learning rate scheduler (linear decay). Evaluation was performed at the end of each epoch. Input text was tokenized using `RobertaTokenizerFast`, and padding was handled by `DataCollatorWithPadding` to ensure dynamic batching. Truncation and padding were enabled during pre-processing to standardize input lengths. No data augmentation or additional pretraining was performed. Performance was evaluated using standard scikit-learn metrics and confusion matrix analysis.

E Entropy-Based IG Threshold Selection

To determine the optimal similarity threshold τ for our entropy-based IG metric (Section 4.2), we swept values from 0.55 to 0.85 in increments of 0.05 as seen in Table 13. For each threshold, we evaluated the predictive utility of IG using 2,000 question-answer pairs via two analyses:

1. **Accelerated Failure Time (AFT) model:** Estimates the effect of IG on the number of turns until game success. Positive coefficients imply that higher IG is associated with *slower* convergence; negative coefficients imply faster convergence.

2. **Spearman rank correlation:** Measures whether higher average IG per game correlates with fewer total questions.

τ	AFT β	AFT p	Spearman ρ	Spearman p
0.55	+0.137	0.423	+0.137	0.343
0.60	**−0.233**	**0.017**	−0.253	0.003
0.65	−0.085	0.507	−0.130	0.369
0.70	−0.057	0.647	−0.222	0.122
0.75	−0.012	0.923	−0.237	0.098
0.80	−0.075	0.560	−0.234	0.103
0.85	+0.060	0.619	**−0.312**	**0.027**

Table 13: AFT model coefficients (mu_ ig_z) and Spearman correlations between entropy-based IG and game length across ConceptNet similarity thresholds τ. Negative AFT coefficients imply faster convergence with higher IG.

Discussion. Threshold $\tau = 0.60$ yields the only statistically significant AFT coefficient ($\beta = -0.233$, $p = 0.017$), suggesting that information gain at this threshold robustly predicts faster convergence. In the AFT model, a negative coefficient indicates that higher IG leads to shorter games. This aligns with the intended role of IG as a proxy for question informativeness.

By contrast, $\tau = 0.85$ achieves the best Spearman correlation ($\rho = 0.312$, $p = 0.027$), but its AFT coefficient is positive and non-significant, suggesting that IG at this threshold may capture broad informativeness trends rather than per-turn utility.

Given these trade-offs, we adopt $\tau = 0.60$ in our main experiments due to its stronger turn-level predictive power. Nonetheless, higher thresholds such as $\tau = 0.8$ may offer value in summarizing informativeness at a more coarse-grained level.

F Bayesian Belief Update

Interpreter. We impose a strictly formatted system instruction (Appendix G) that requests a comma-separated list of at most five *concept*:score pairs, where every score lies in the open interval $(-1, 1)$. Gemini is queried with a low-temperature $n$ucleus configuration (temperature 0.3, $t_p = 0.8$) to obtain deterministic extractions. This is transformed into a normalized dictionary $\tilde{s}(y) \in [0, 1]$ that can be consumed by the Bayesian update.

Soft-evidence Belief Update. Let $b_t(y)$ be the categorical belief over candidate concepts at turn t, and let $s_t(y) \in (0, 1)$ be the Interpreter's normalized relevance score for concept y. We apply a multiplicative soft-evidence rule (1) where $\alpha = 1$ controls update strength and $\varepsilon = 10^{-12}$ prevents zero probabilities.

Tuning Soft-evidence Scale α. We performed a small grid-search over the scaling constant α in Eq. (1) using a held-out sample of 40 games. After each game we computed the Spearman correlation between information gain from each question and the number of turns remaining. Table 14 summarizes the outcome.

All three scales yield a significant negative correlation between average IG and dialogue length, but the strength of the relationship varies:

At $\alpha = 0.5$ the update is conservative, producing a moderate correlation ($r = -0.55$, $p = 2.4 \times 10^{-4}$).

Increasing the weight to $\alpha = 1.0$ strengthens the link ($r = -0.76$, $p = 1.4 \times 10^{-8}$), indicating that a unit-scale multiplier best aligns information gain with faster convergence.

Pushing to $\alpha = 2.0$ causes the correlation to slip back to $r = -0.47$ ($p = 2.1 \times 10^{-3}$), suggesting mild over-confidence that slightly blunts the predictive value of IG.

We therefore fix $\alpha = 1$ in all subsequent experiments as it provides the best trade-off between statistical significance and stability.

α	Spearman r (IG vs. turns)	p-value
0.5	-0.55	2.40×10^{-4}
1.0	-0.76	1.40×10^{-8}
2.0	-0.47	2.10×10^{-3}

Table 14: Effect of the multiplicative scale α on the correlation between Average information gain per question and dialogue length ($N = 40$ games).

$\alpha = 0.5$		
Threshold	ρ	p
none	-0.62	2.40×10^{-5}
15%	$\mathbf{-0.75}$	$\mathbf{2.2\times10^{-8}}$
25%	-0.55	2.40×10^{-4}
35%	-0.29	7.20×10^{-2}
45%	$+0.15$	3.60×10^{-1}
55%	$+0.49$	1.30×10^{-3}
65%	$+0.49$	1.20×10^{-3}
$\alpha = 1$		
Threshold	ρ	p
none	-0.63	1.30×10^{-5}
15%	-0.31	5.30×10^{-2}
25%	-0.76	1.40×10^{-8}
35%	$\mathbf{-0.78}$	$\mathbf{2.40\times10^{-9}}$
45%	-0.10	5.40×10^{-1}
55%	$+0.18$	2.70×10^{-1}
65%	$+0.45$	3.90×10^{-3}
$\alpha = 2$		
Threshold	ρ	p
none	-0.30	6.30×10^{-2}
15%	$+0.10$	5.20×10^{-1}
25%	-0.47	2.20×10^{-3}
35%	$\mathbf{-0.61}$	$\mathbf{2.60\times10^{-5}}$
45%	$+0.56$	2.00×10^{-4}
55%	$+0.71$	3.70×10^{-7}
65%	$+0.74$	3.70×10^{-8}

Table 15: Spearman correlation (ρ) between average information gain and dialogue length under different pruning thresholds. Bold numbers mark the strongest *negative* correlation for each α (N=40 games).

Effect of the pruning threshold. We evaluated a grid of soft-evidence scales $\alpha \in \{0.5, 1, 2\}$ and pruning thresholds $\{$ none, 15%, 25%, 35%, 45%, 55%, 65%$\}$ (see Table 15). For the conservative weight ($\alpha = 0.5$) the IG-turn correlation peaks at a strong

$\rho = -0.75$ with a 15 % threshold, but weakens rapidly, and even changes sign, as the threshold increases. The aggressive setting ($\alpha = 2$) shows the opposite pattern: it still yields a notable $\rho = -0.61$ at 35 %, yet flips to large *positive* correlations ($\rho > +0.55$) when more than 45 % of the belief mass is pruned, indicating over-confident collapse. By contrast, the mid-range scale $\alpha = 1$ is consistently stable, producing highly significant negative correlations across the 25-35 % band (strongest at 35 %, $\rho \approx -0.78$, $p < 2.4 \times 10^{-9}$). Balancing robustness with predictive strength, we therefore adopt $\alpha = 1$ and a 35 % pruning threshold in all subsequent experiments.

G Prompts

- **Guesser Any Question Prompt:** "You are named Guesser. You are trying to guess what physical object or physical material Oracle is thinking of. You will ask questions to get more information about the object. Open ended question are questions that have a large variety of answers. A question is not open ended if the only answers to it are yes or no. An example of an open ended question is: What material is the object made of? An example of a not open ended question is: Is the material the object is made of copper? You may ask open or not open questions. You can make direct guesses on what the object is. When you believe you have enough information about the object you will guess what it is. Your guessing will be informed by your prior guesses. Do not ask the question: What is the object? Do not explain your reasoning in your guess, only say your question. You will start each message with Guesser said: . If you guess wrong you will ask more questions about the object until you have enough information to guess again."

- **Guesser Open Question Prompt:** "You are named Guesser. You are trying to guess what physical object or physical material Oracle is thinking of. You will ask open ended questions to get more information about the object. Open ended question are questions that have a large variety of answers. A question is not open ended if the only answers to it are yes or no. An example of an open ended question is: What material is the object made of? An example of a not open ended question is: Is the material the object is made of copper? You can make direct guesses on what the object is. When you believe you have enough information about the object you will guess what it is. Your guessing will be informed by your prior guesses. Do not explain your reasoning in your guess, only say your question. You will start each message with Guesser said: . If you guess wrong you will ask more questions about the object until you have enough information to guess again. "

- **Oracle Prompt:** "You are named Oracle. Guesser is trying to guess what physical object you are thinking of. When Guesser correctly guesses the object, you will only return Correct. If Guesser asks: is it a type of object, and the object is the same as your object then this is also a correct guess. You can not make any guesses or ask any questions. You start each response with Oracle said: . The object you are thinking of is a "

- **Checker Prompt:** "You are an expert annotator that is categorizing the questions asked by Guesser in an object guessing game. There are 5 types of questions. The first type are Attribute questions, these involve the physical attributes of the physical object. Examples of Attribute questions are: Is the object made of metal? What color is the object? What shape is the object? The second type of questions are Function questions, these involve the function of the physical object. Example of Function questions are: Is the object used for communication? Is the object used for building? Is the object used for eating food? The third type of questions are Location questions, these ask about where a physical object is located. Examples of Location questions are: Is the object in the bedroom? Is the object located inside or outside? Is the object on the desk? The fourth type of questions are Category questions, these ask if the physical object belong to certain category of objects. Examples of Category questions are: Is the object a type of car? If the object a type of furniture? The fifth type of questions are Direct questions, these are questions that directly guess what the object is. Examples of Direct questions are: Is the object a phone? Is the object a bed? Is the object a knife? After be-

ing given Guesser's question return only what type of question it is. Return only one of the following 5 words: Attribute, Function, Location, Category, or Direct, based on what type of question Guesser is asking. Do not explain your reasoning or your thinking. What type of question is Guesser asking? "

- **Interpreter Prompt:** "You are named the Interpreter. Your task is to generate a comma-separated relevance-scored list of candidate concepts based on the Guesser's questions and the Oracle's answers to that question. Candidate concepts are inferences you can make about the physical or functional attributes or location or category of the object that the Oracle is answering about. Rules 1. Every concept and its corresponding score must be separated by a colon and each concept-score pair must followed by a comma 2. Each score is a float in (-1, 1). 1 = strongly positive correlation, -1 = strongly negative correlation. 3. Do not output any additional text, explanation, punctuation (except commas), or commentary, metadata tags, special tokens, statements, explanations, additional works, questions or guesses."