AFFORDANCE20Q: Evaluating Affordance Reasoning from Physical Properties

Yifan Jiang[1], **Meige Yang**[2], **Zitong Li**[2], **Jay Pujara**[1]
[1]Information Sciences Institute, University of Southern California
[2]University of Southern California
{yifjia,jpujara}@isi.edu, maggieya@usc.edu, alex.zitong.li@gmail.com

Abstract

Affordance reasoning, the inference of an object's action possibilities from its physical properties (e.g., shape and material), is fundamental to human physical understanding and increasingly critical for Large Language Models (LLMs). However, existing affordance benchmarks largely expose explicit object identities in the evaluation setup, allowing models to rely on memorized object-affordance mappings rather than reasoning over physical properties. To address this gap, we introduce AFFORDANCE20Q, a novel affordance reasoning benchmark formulated as a 20-Questions game without exposing the object's identity. In each game, the model identifies a hidden object's affordance from a candidate set by asking yes/no questions about its physical properties. AFFORDANCE20Q comprises 1,009 games over 454 objects and 59 affordances, all manually filtered, refined, and annotated. We conduct comprehensive experiments with 15 state-of-the-art LLMs and find a substantial gap (∼20 points) compared to human performance. A KL-based information-gain (IG) analysis further shows that models fail to ask discriminating questions as the game progresses. To close the gap, we develop KB-Anchored Rule Induction (KARI), a pipeline based on LLMs that generates affordance rules grounded in evidence from knowledge bases (KBs). KARI improves open-source LLMs by up to 15.2 points, while the limited coverage of KBs hinders further gains. We release all our code and data at https://github.com/1171-jpg/Affordance20Q.git.

1 Introduction

When humans encounter an object, they reason about what actions it supports directly from its physical properties, including its shape, material, and structure. Gibson (1977) formalized this capacity as **affordance reasoning** and placed it at the core of physical understanding. Affordance reasoning operates on physical properties that every object exhibits, and therefore supports a wide range of physical interaction, from fluently using familiar everyday objects (Norman, 2013) to robustly handle novel ones or creatively repurposing familiar ones beyond their typical use (Duncker and Lees, 1945; German and Defeyter, 2000). Such reasoning is increasingly crucial for Large Language Models (LLMs) as recent progress has led to their integration into daily human activities (Yang et al., 2025; Singh et al., 2025), and especially for the growing class of LLM-driven embodied systems (Driess et al., 2023; Zhang et al., 2025) that aim for physical application. LLMs are expected not only to understand everyday tool use (Wang et al., 2023b, 2026b), but also to generalize this capability to unfamiliar objects or scenarios beyond their training experience (Tian et al., 2024; Jiang et al., 2023b). For example, in embodied robotic manipulation, a model should not only recognize that a knife affords cutting, but also reason that any object with a sharp rigid edge affords the same action (Tang et al., 2025; Xu et al., 2022).

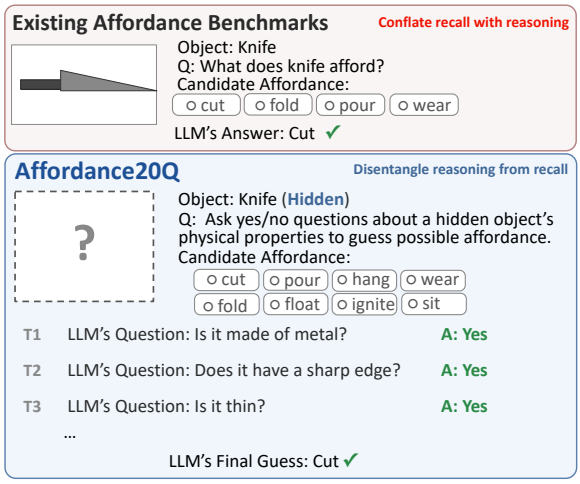


Figure 1: Comparison between existing affordance benchmarks (top) and AFFORDANCE20Q (bottom).

Recognizing its importance, recent research has

Findings of the Association for Computational Linguistics: EMNLP 2026, pages 35930–35948
October 24-29, 2026 ©2026 Association for Computational Linguistics

introduced diverse benchmarks for affordance reasoning, spanning multiple modalities (Wang et al., 2026b; Yu et al., 2025) and task types (Qasemi et al., 2022; Tian et al., 2024). Yet all setups make the object's identity explicit by giving the class or object name (Figure 1), conflating recall of object-affordance mappings with reasoning from physical properties. A model knowing that the object is a knife can answer "it affords cutting" simply by recalling a stored mapping (Persiani and Hellström, 2019), without ever consulting its rigid, sharp-edged physical properties. Such conflation prevents existing benchmarks from accurately measuring affordance reasoning capability. Recall-based models can pass these benchmarks within the training distribution, but fail on the unseen objects and atypical uses common in real-world physical interaction (Gjerde et al., 2025; Wu et al., 2024).

To disentangle reasoning from recall, we introduce AFFORDANCE20Q, a benchmark cast in a 20-Questions (20Q) game setting (Bruner et al., 1966), in which models identify an object's affordance without knowing its identity. As shown in Figure 1, in each game, given candidate affordances, the model narrows the candidates through multi-turn yes/no questions about the hidden object's physical properties, such as material and shape. The exclusion of object identity requires the model to identify the correct affordance by reasoning from physical evidence rather than recalling stored object-affordance mappings (Hutson et al., 2025). To construct AFFORDANCE20Q, we use a three-stage pipeline that first collects physical objects and affordances from the existing corpus (Jiang and Riloff, 2021) and commonsense knowledge base (Ilievski et al., 2021), then enriches them via LLM generation, and finally manually filters, refines, and annotates (object, affordance) pairs. In the end, AFFORDANCE20Q consists of 1,009 games covering 454 objects and 59 affordances.

With AFFORDANCE20Q, we evaluate 15 state-of-the-art LLMs with different sizes and architectures, and find a substantial gap (∼20 points) compared to human performance, with the strongest model reaching only 45.9%. To further diagnose the gap, we trace each game with a KL-based information-gain (IG) metric and find that models repeatedly ask low-IG questions that fail to narrow the candidate set across turns. To close this gap, we develop KB-Anchored Rule Induction (KARI), a pipeline that uses knowledge bases to both inspire LLM rule generation and re-ground the generated rules through post-hoc validation, ensuring that generated rules remain anchored in physical commonsense rather than free-form LLM speculation. KARI improves open-source LLMs by up to 15.2 points, partially closing the gap, while the remaining shortfall traces to the coverage limits of current commonsense knowledge bases.

We summarize our contributions as follows: 1) We introduce AFFORDANCE20Q, a 20-Questions benchmark that tests affordance reasoning from physical properties rather than object-identity recall, comprising 1,009 games over 454 objects and 59 affordances. 2) We conduct comprehensive experiments with 15 state-of-the-art LLMs, revealing a substantial gap to human performance, with information-gain analysis showing that models fail to ask discriminating questions as the game progresses. 3) We develop KB-Anchored Rule Induction (KARI), a pipeline that combines LLMs and knowledge bases, improving open-source LLMs by up to 15.2 points. We release all code and data.

2 Related Work

Affordance Reasoning Benchmarks The importance of affordance reasoning (Gibson, 1977) has motivated a wide range of benchmarks across diverse input formats and modalities. In the vision domain, early work focuses on grounding part/object-level affordance in either 3D shape and part geometry (Deng et al., 2021; Xu et al., 2022) or 2D image input (Nguyen et al., 2017; Luo et al., 2022; Li et al., 2023). Recent work shifts to evaluating models' affordance reasoning with both image inputs and text instructions across different task setups (Wang et al., 2026b; Yu et al., 2025; Wang et al., 2026a; Zhu et al., 2025; Wan et al., 2025). In the text domain, parallel work formats affordance reasoning as question-answering tasks with object names or descriptions provided as context (Bisk et al., 2020; Aroca-Ouellette et al., 2021; Wang et al., 2023b; Adak et al., 2024; Gjerde et al., 2025). Although a few works (Li et al., 2023; Gjerde et al., 2025) format affordance reasoning based on physical properties, none of them remove object identity to prevent recall of object-affordance mappings. In contrast, AFFORDANCE20Q is the first benchmark to exclude object identity from the input and require the model to infer affordance from physical properties through multi-turn questioning, disentangling reasoning from recall.

20-Questions Games and Active Question-Asking The 20-Questions (20Q) game was first used in cognitive science to study information-seeking behavior (Bruner et al., 1966; Ruggeri et al., 2016). In each game, a questioner aims to identify a hidden target through a sequence of yes/no questions within a fixed number of turns. As active question-asking ability becomes increasingly important in real-world human-computer interaction scenarios (e.g., task disambiguation (Kobalczyk et al., 2025), medical diagnosis (Li et al., 2024)), the 20Q game has recently been widely adopted to analyze LLMs' multi-turn reasoning and information-seeking abilities (Bertolazzi et al., 2023; Hutson et al., 2025; Zhang et al., 2024; Mazzaccara et al., 2024). However, all current work chooses the object or entity as the candidate space. For example, Zhang et al. (2024) and Hutson et al. (2025) require LLMs to strategically ask questions to identify a hidden target object. AFFORDANCE20Q is the first to adopt the 20Q game for affordance reasoning, which also aligns with Gibson's active-perception view of affordance (Gibson, 1977). We further introduce a novel KL-based information-gain metric to evaluate model question effectiveness across turns.

3 AFFORDANCE20Q Construction

In this section, we first formalize the game setup of AFFORDANCE20Q (§3.1), then describe the design of our three-stage collection pipeline (§3.2), and end up with the data statistics (§3.3).

3.1 Game Formulation

Following recent 20Q adaptations for LLM evaluation (Hutson et al., 2025; Zhang et al., 2024), each game in AFFORDANCE20Q is defined by a hidden target object o^* and a candidate affordance set $\mathcal{A} = \{a^*, a_1, \ldots, a_7\}$ of 8 affordances, in which the target a^* is an affordance that o^* possesses and the remaining 7 are distractors. Three agents participate: a *Questioner* that observes only $\mathcal{A}$ and asks yes/no questions (q) about o^*'s physical properties to identify a^*, a *Checker* that ensures each question is both well-formed and grounded in one of the physical-property dimensions (e.g., material, shape), preventing information leakage, and an *Oracle* that has access to o^* and provides answers (r) to physical-property questions. At each turn t, the Questioner produces a question q_t based on the dialogue history $H_{t-1} = \{(q_1, r_1), \ldots, (q_{t-1}, r_{t-1})\}$, the Checker validates q_t, and the Oracle returns an answer r_t. The game succeeds if the Questioner correctly identifies a^* within a budget of $T = 20$ turns. It fails if the Questioner makes an incorrect guess or exhausts the turn budget.

3.2 Three-Stage Collection Pipeline

Unlike previous 20Q setups (Hutson et al., 2025; Zhang et al., 2024) that frame the task as identifying a hidden object, to evaluate affordance reasoning, AFFORDANCE20Q frames it as identifying a hidden object's affordance, where the hidden object remains but the Questioner reasons over its physical properties. To ensure this reasoning chain is valid, AFFORDANCE20Q excludes any object-affordance pair whose affordance is not deducible from physical-property dimensions alone (e.g., a microwave's heating affordance comes from its magnetron). Given the cost of fully manual curation, we design a semi-automatic three-stage collection pipeline to construct AFFORDANCE20Q in a scalable way.

Stage 1: Initial Object and Affordance Collection. Our initial object pool is the list of human-made physical objects introduced in (Jiang and Riloff, 2021). We build the initial affordance pool by querying Commonsense Knowledge Graph (CSKG) (Ilievski et al., 2021) for the relations describing what objects are capable of and used for (e.g., `CapableOf`, `UsedFor`). CSKG integrates seven commonsense knowledge bases (e.g., ConceptNet (Speer et al., 2017), WordNet (Fellbaum, 2010)) into one schema and thus offers denser affordance coverage than any single source.

Stage 2: Affordance Expansion and Pool Filtering. To further expand the affordance coverage, we query CSKG for each object's physical properties and prompt an LLM to propose additional candidate affordances conditioned on them. To remove entries unsuitable for our task, we then manually filter both pools based on specific principles. For affordances, we remove entries that violate the following three dimensions: (1) *context-dependence*, the affordance depends on external context rather than the object itself (e.g., being a gift), (2) *mechanism-dependence*, the affordance depends on hidden internal mechanisms (e.g., playing video), and (3) *over-generality*, the affordance is too abstract to be discriminative (e.g., useful). For objects, we drop entries that are not discrete physical objects (e.g., raw material, human body part) or whose function depends on hidden internal

mechanisms (e.g., microwave, TV). After filtering, we refine affordance names and write a short description for each, and consolidate each object's CSKG properties by removing irrelevant or conflicting entries, forming each object's *property set* for Stage 3 (Details in Appendix E).

Stage 3: Annotation and Object Specialization. In this stage, we annotate every (object, affordance) pair, determining whether each affordance applies to its paired object. Additionally, as each object's *property set* from CSKG tends to be generic and lacks specific physical details (e.g., some spatulas have an internal hollow bar that filters water), we also specialize each object's property set during annotation. We first prompt an LLM to label every (object, affordance) pair as *YES*, *NO*, or *MAYBE* given the object's property set and the affordance's definition, naming the additional property required for each *MAYBE* label. We then manually verify each *YES*/*NO* label against the object's property set, and for each *MAYBE*, decide whether the proposed property applies, and if it does, add the property to the set, turn the label to *YES*, and re-check consistency.

3.3 Implementation Details and Data Statistics

We run the pipeline with GPT-4.1 (Achiam et al., 2023) as the main LLM due to its promising performance in public benchmarks (Liang et al., 2022). Six human annotators were involved in the manual annotation and refinement process in Stages 2 and 3. We further recruited three additional annotators to re-annotate a random subset of 1,298 pairs, achieving an 85.2% majority agreement with our released labels (Fleiss $\kappa = 0.82$ (Fleiss, 1971)). Among the three annotators themselves, pairwise agreement averages 90.9%, and all three agree on 86.4% of the pairs. Applying the three-stage pipeline yields 454 objects, each with a property set, 59 affordances, and a label for every (object, affordance) pair. We sample 1,009 (object, target affordance) game instances as the test set, balanced across target affordances so that no single affordance dominates, forming AFFORDANCE20Q.

4 KB-Anchored Rule Induction

Prior work on affordance and commonsense reasoning either distills rules from KBs bound by a fixed schema (Zhu et al., 2014), or generates them with LLMs but risks hallucination (West et al., 2022). We investigate whether combining KB knowledge with LLM reasoning yields rules that are both grounded and expressive, thereby assisting the Questioner in our task. Therefore, we develop KB-Anchored Rule Induction (KARI), a pipeline that uses an LLM to induce a compositional rule per affordance grounded in evidence from external KBs. We next describe KARI's rule format and how its components combine KB knowledge with LLM reasoning during rule generation.

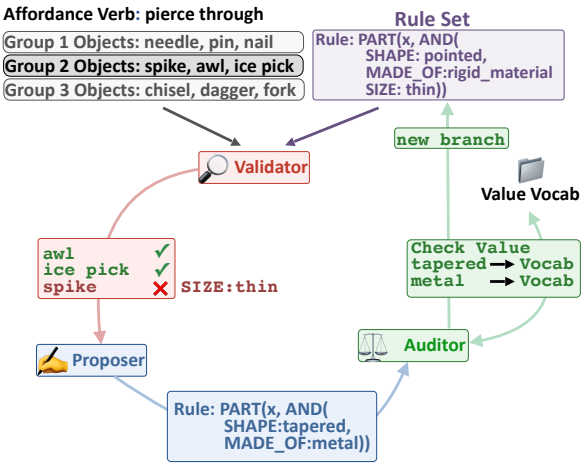


Figure 2: Illustration of KB-Anchored Rule Induction (KARI)

4.1 Rule Format

A KARI rule is a tree expression stating the static physical conditions enabling an affordance. KARI's rule grammar has seven operators in three categories. (i) Four atomic **predicates** describe a physical property of the entity: `MADE_OF`, `SHAPE`, `SIZE`, and `SURFACE`. Each takes a **value** to represent one physical property. For instance, `MADE_OF: metal` asserts that the entity is made of metal. (ii) Two **combinators** produce a **combination** of multiple atomic predicates: `AND` and `OR`, which can nest inside each other. For instance, `AND(MADE_OF: metal, SHAPE: pointed)` requires the entity to be both metal and pointed. (iii) The constructor `PART(x, predicate/combination)` represents a specific part of the object, where `x` is a variable that can refer to any part. We introduce `PART` because many affordances can be enabled by one specific part rather than the entire object, and we use additional part variables (e.g., `y`, `z`) when multiple distinct parts are required.

4.2 Rule Proposer, Validator, and Auditor

KARI consists of three components, each for a specific purpose (Figure 2). We first introduce each component, then illustrate the overall KARI pipeline. We collect affordance verbs[1] and corresponding positive objects together with their materials, parts, and other physical properties from CSKG (Ilievski et al., 2021) and the Aristo Tuple KB (Dalvi et al., 2017) as our initial data.

Proposer. The proposer is the only component that writes rule expressions. Given an affordance verb and a set of objects with their physical properties from a knowledge base, it either drafts a new rule or extends an existing one by appending a new `OR` branch. We restrict the proposer's behavior to be add-only to ensure possible errors will only occur in the new branch.

Validator. The validator takes an object and a set of rules as input. Since the object is a positive example of the affordance verb according to the KB, each rule is expected to evaluate true on it. For each rule, an LLM labels every atom as *YES* or *NO*, and a code script evaluates the `AND`/`OR` tree to determine whether the rule covers the object.

Auditor. The auditor validates each value the proposer introduces and maintains a vocabulary of accepted values per predicate, which prevents ill-typed values from entering the rule and the vocabulary from drifting through redundant synonyms. Given a rule, the auditor checks whether each value is valid for its predicate (e.g., `MADE_OF: cylinder` is rejected), then either maps the value to an existing synonym in the vocabulary or adds it as a new entry for future use.

4.3 Pipeline

Figure 2 shows the overview of the pipeline. We split the collected data into three groups and call each component in a loop. In round 1, the proposer drafts a seed rule on the first group of data, and the auditor updates the values and the vocabulary. In rounds 2 and 3, the validator first validates the previous rule, and the proposer proposes a new rule for uncovered objects in the new group, with the auditor performing the same update again.

Adapting KARI to our collected data with QWEN3-14B for all components produces 2,223 rules. In the inference time, for each game, we compute the sentence similarity (Ni et al., 2022) between each candidate affordance and every generated rule's affordance verb, and match the candidate to its highest-scoring rule when the similarity exceeds 0.7. Matched rules are verbalized at the end of the Questioner's system prompt. A candidate affordance with no match above the threshold receives no rule. Full details and rule examples are presented in Appendix A.

[1]We use *affordance verb* to refer to affordances collected from KBs in the KARI pipeline.

5 Experiment Setup

We evaluate on the 1,009-game test split of AFFORDANCE20Q (Section 3), with the human baseline run on a sampled subset. The Questioner, Oracle, and Checker all run at temperature 0. Full prompts and other details are in Appendix C.

5.1 Questioners

To ensure a comprehensive evaluation, we test 15 LLMs as Questioners, grouped into open-source and closed-source models.

Open-source LLMs. We evaluate ten open-source models spanning a wide range of scales. Eight are dense models between 8B and 14B parameters: QWEN3-8/14B and QWEN3.5-9B (Yang et al., 2025), PHI-4-14B (Abdin et al., 2024), LLAMA-3.1-8B (Grattafiori et al., 2024), MINISTRAL-8B (Liu et al., 2026a), NEMOTRON-9B (Blakeman et al., 2025), and GEMMA-3-12B (Team et al., 2025). The other two are the mixture-of-experts models DEEPSEEK-V4-PRO and DEEPSEEK-V4-FLASH (DeepSeek-AI, 2026).

Closed-source LLMs. We evaluate five proprietary models accessed through their official APIs: GPT-5 and GPT-5-MINI (Singh et al., 2025), GEMINI-2.5-PRO and GEMINI-2.5-FLASH (Comanici et al., 2025), and MINIMAX-M2.5 (MiniMax, 2025).

Reference points. 1) **HUMAN**: five annotators play a 30% subset under the same game rules, and we report the average result. We further include two non-LLM reference points to show the possible performance range. To enable the final guess, for each candidate affordance, we select ten representative objects, and use the fraction of them that remain consistent with all questions and their corresponding Oracle answers. We then softmax these fractions across affordances as the likelihood of being the target. 2) **FIX20Q** asks the 20 most frequent questions from the LLM evaluation in a fixed order (Appendix C.1), then samples its final

guess from a softmax over these likelihoods, and we report the average over five runs. 3) **OPTIMAL** is an information-theoretic upper bound with full access to all candidates' representative objects. At every turn, it asks the question that drives the target affordance's likelihood as high as possible, and gives the final answer once a single affordance's likelihood exceeds a specific threshold (0.9).

We also test ENTROPY-MAX, which asks the question that maximally reduces the entropy over the eight candidates without knowing the target. It reaches 51.2% in 4.6 turns, showing that narrowing the search space alone is not enough for success (details in Appendix C.2).

5.2 Oracle and Checker

We use a single QWEN3-14B instance for both the Oracle and the Checker. Since we provide each object's *property set* (Section 3.2) as context, the Oracle agrees with human labels on 93.4% of 300 manually verified questions (MATERIAL 98.0%, SHAPE 95.7%, SURFACE 88.2%, SIZE 84.8%), with errors concentrated in borderline size comparisons (Appendix C.5). We also test a stronger GPT-5 Oracle on six Questioners, which shifts success rates by at most 2.6 points and leaves the ranking unchanged (Appendix C.5).

5.3 Metrics

We report three metrics: 1) **Success Rate (SR)**, the fraction of games in which the target affordance is identified within the 20-turn budget, 2) **Turns**, the average number of turns in solved games, and 3) **Information Gain (IG)**, a per-turn measure of how fast each question narrows the candidate affordances. Using the same representative objects, let $n_t(a)$ be the number of affordance a's representative objects still consistent with the dialogue history after turn t, giving a distribution $b_t(a) = n_t(a)/\sum_{a'} n_t(a')$ over the 8 candidates. IG is the KL divergence between consecutive distributions,

$$\mathrm{IG}_t = D_{\mathrm{KL}}(b_t \,\|\, b_{t-1}) = \sum_a b_t(a)\, \log_2 \frac{b_t(a)}{b_{t-1}(a)}. \quad (1)$$

6 Results

We focus on five research questions: *1) How well do LLMs reason about affordances from physical properties? 2) How do LLMs behave when reasoning about different affordances? 3) How effectively do LLMs gather information through questioning? 4) Can KARI's rules improve LLMs' affordance reasoning? 5) What are the typical success and failure modes?*

6.1 Main Results

We report the success rate and average turns in Table 1. OPTIMAL reaches 100% in 2.5 turns, while FIX20Q reaches only 24.8% despite using the full 20-question budget, showing that extensive questioning without affordance reasoning specific to each game is not beneficial for success. Humans reach 64.2% in 10.7 turns, and their gap to OPTIMAL shows the challenging nature of AFFORDANCE20Q. We note that 2.5 turns is a theoretical minimum, achievable only by optimally ordering questions to rule out distractor affordances or confirm the target affordance, whereas humans can ultimately solve the games in more turns, confirming the validity of AFFORDANCE20Q.

For both open- and closed-source LLMs, **all models fall far short of human performance, with gaps ranging from roughly 20 to 50 points**. Most open-source LLMs perform similarly to FIX20Q, ranging from 14.9% to 27.4%, showing they struggle to solve each game with appropriate affordance reasoning. DEEPSEEK-V4-FLASH and DEEPSEEK-V4-PRO are the clear exceptions, consistent with their strong results on other benchmarks (DeepSeek-AI, 2026) and likely aided by their mixture-of-experts architecture. All closed-source LLMs except GPT-5-MINI achieve higher success rates than the 8B to 14B models. Notably, GEMINI-2.5-PRO outperforms the strongest open-source LLM (DEEPSEEK-V4-FLASH) by 4.6 points, showing a 4.6-point gap between the two groups' best models. Models also differ in their turn usage, which does not guarantee a higher success rate. For example, NEMOTRON-9B and PHI-4-14B tend to end games early, while QWEN3.5-9B tends to exhaust its turn budget, but neither yields a higher success rate.

6.2 Affordance Difficulty

We further break down model performance by affordance (full results in Appendix D), which ranges widely from 3.1% to 56.0%. In general, affordances that can be inferred from a single physical property tend to be easy, such as *transmit_light* (51.4%), established once the object is

Questioner	Vanilla SR	Vanilla Turns	+KARI SR	+KARI Turns
Reference Points				
OPTIMAL	100.0	2.5	–	–
HUMAN	64.2±7.3	10.7±1.3	–	–
FIX20Q	24.8±0.9	20.0	–	–
Open-source LLMs				
LLAMA-3.1-8B	18.4	14.5	31.3$_{\uparrow 12.9}$	20.0
MINISTRAL-8B	17.7	15.1	28.0$_{\uparrow 10.3}$	12.6
QWEN3-8B	26.3	16.1	33.4$_{\uparrow 7.1}$	11.7
NEMOTRON-9B	14.9	8.6	30.0$_{\uparrow 15.1}$	17.4
QWEN3.5-9B	25.1	18.6	33.2$_{\uparrow 8.1}$	14.3
GEMMA-3-12B	27.4	15.1	33.7$_{\uparrow 6.3}$	18.6
QWEN3-14B	27.3	18.6	32.9$_{\uparrow 5.6}$	10.9
PHI-4-14B	26.4	11.2	30.5$_{\uparrow 4.1}$	11.9
DEEPSEEK-V4-FLASH	41.3	16.9	37.9$_{\downarrow 3.4}$	10.6
DEEPSEEK-V4-PRO	40.5	18.8	38.5$_{\downarrow 2.0}$	11.2
Closed-source LLMs				
GPT-5-MINI	22.8	16.9	31.2$_{\uparrow 8.4}$	10.9
GPT-5	35.5	14.5	32.4$_{\downarrow 3.1}$	7.0
MINIMAX-M2.5	34.9	17.2	35.5$_{\uparrow 0.6}$	9.3
GEMINI-2.5-FLASH	41.4	17.3	33.4$_{\downarrow 8.0}$	9.9
GEMINI-2.5-PRO	**45.9**	18.5	**40.0**$_{\downarrow 5.9}$	12.1

Table 1: Main results on AFFORDANCE20Q, with and without KARI's rule integration. SR is the success rate (%), and Turns is the average turns. The best overall result is in **bold**, the best in each category underlined.

known to be transparent, or *conduct_heat* (55.2%), once it is known to be made of metal material. The hardest affordances instead require reasoning over multiple physical properties or alternative affordance rules, such as *sink_in_water* (3.8%), which jointly depends on the material of the object, whether it contains a hollow structure inside, and whether the material can absorb the water, or *hang_from_above* (10.6%), where the object has a hook-shaped, ring-shaped, or strap-like part that can support this affordance. We next compare open- and closed-source LLMs' performance on each affordance. Open-source LLMs often perform better when an affordance hinges on a single physical property, while closed-source LLMs spend more turns yet result in a lower success rate, suggesting a potential overthinking behavior. On the other hand, closed-source LLMs outperform open-source LLMs on affordances requiring multiple physical properties, with similar turn usage, highlighting their stronger reasoning ability.

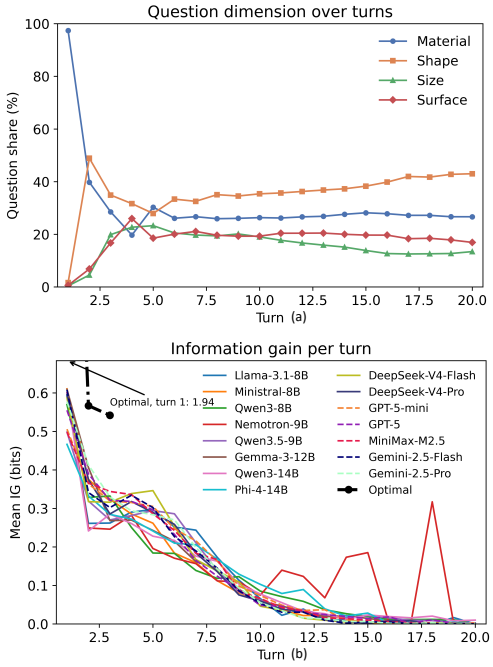


Figure 3: Question behavior (a) and information gain (b) over turns.

6.3 Question behavior and Information Gain

We use the four atomic predicates in our KARI rule system to categorize LLM questions into different physical-property dimensions and visualize their share at each turn in Figure 3(a). We observe a clear trend: LLMs prefer to start the game with a material question and later adopt shape questions to guide their decisions. Size and surface questions occupy only a small share in the initial stage and grow afterward, while shape questions come to dominate the share (around 40%). We further examine whether this question behavior is meaningful by scoring each turn with its information gain (IG) (Figure 3(b)). We find that material and shape questions provide useful information gain in the initial turns (1 to 5), although the large gap

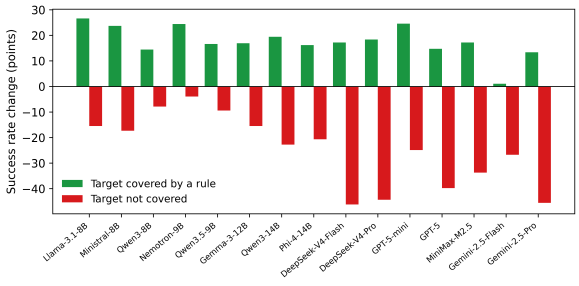


Figure 4: Accuracy change split by the target affordance coverage.

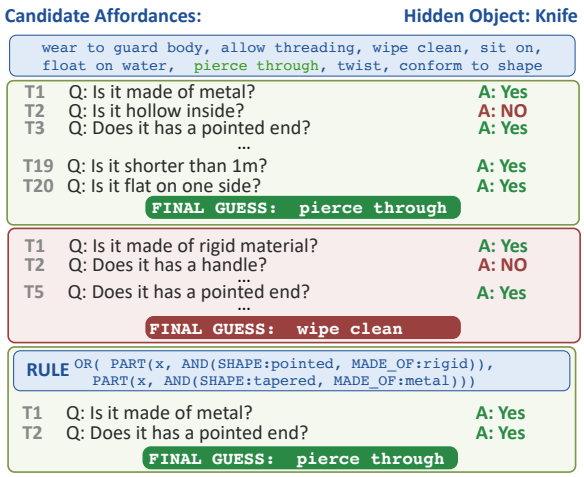


Figure 5: Success and failure games with the target affordance *pierce through* (hidden object: knife).

from OPTIMAL shows these are still not the best questions to ask. The IG then collapses after turn 5 and gradually approaches zero, even though models keep asking questions from different categories. This shows that, in the early turns, LLMs can ask basic questions that roughly build up likelihoods over the candidate affordances to assist affordance reasoning. As the game proceeds, they cannot ask more discriminating questions needed to separate the target affordance from the remaining distractors, leaving them uncertain at the final guess. This also confirms our earlier observation that asking more questions does not improve the success rate.

6.4 KARI's Rule Integration

The result of KARI's rule integration (Table 1) is mainly two-fold. For models with 8B to 14B parameters, we observe a clear improvement in success rate, ranging from 4.1 to 15.2 points, verifying the effectiveness of the generated rules. We also notice that the rules have a mixed impact on the turn number for these models: a huge decrease is observed for QWEN3-8B/14B, while NEMOTRON-9B doubles its turns, and LLAMA-3.1-8B even uses all 20 turns. For most closed-source LLMs and DeepSeek-V4 series, we find that KARI's rules bring a drop in both success rate and turns, suggesting the rules lead these LLMs to make fast but unsure final guesses. KARI's rules are generated from knowledge bases whose limited coverage is a known limitation in prior work (Bosselut et al., 2019; Hwang et al., 2021). We note that only 67.4% of the games in AFFORDANCE20Q have their target affordance covered by a KARI rule. We split games by this coverage to analyze whether the limited coverage is a barrier to the improvement brought by KARI's rules (Figure 4). On covered target affordances, KARI's rules actually supply useful affordance knowledge and improve every LLM by 17.7 points on average. On uncovered ones, the distractor affordances' rules mislead the LLM toward irrelevant properties and lower the success rate by 25.0 points, hurting the strongest LLMs most, which can already solve some of these games even without rules (further analysis in Appendix B).

6.5 Case Study

We present three games with the same game setting in Figure 5. The first two use DEEPSEEK-V4-FLASH and MINIMAX-M2.5 as the Questioner; the third reuses DEEPSEEK-V4-FLASH assisted with KARI's rules. [2] In the first example, the model has already collected enough physical properties to support the target affordance *pierce through* early in the game (T1, T3), yet it continues asking until the turn budget is exhausted. Although it eventually produces the correct final guess, the behavior suggests overthinking or low confidence in reasoning from physical properties to a specific affordance. The second example further illustrates this pattern: the model gathers sufficient evidence for the target affordance (T1, T5) but still arrives at a wrong final guess. In contrast, the third example shows that KARI's rule can provide explicit and effective affordance knowledge, allowing the same model to solve the game accurately in just two turns.

7 Benchmark Validity

The results above show a large gap between LLMs and humans. We further verify that this gap re-

[2] In the actual game, rules mapped to the other candidate affordances are also presented. We omit them here for space.

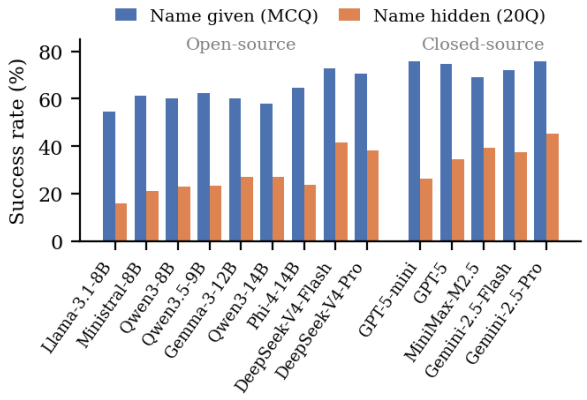


Figure 6: Success rate on the same 295 games with the object name given (single-turn multiple choice) versus hidden (20Q).

flects affordance reasoning from physical properties rather than recall of object identity.

Existence of the recall shortcut. We convert the games into single-turn multiple-choice questions where the object name is directly given and the model picks one of the 8 candidate affordances (5 games per target affordance, 295 games in total). As shown in Figure 6, all models perform reasonably well (54.6% to 75.6%) once the name is given, far above their success rates on the same games with the name hidden. Models can answer from the object name alone, without exploring the object's physical properties, which is exactly the shortcut our hidden-identity design cuts off.

Identity remains hidden during the game. A remaining concern is that models might still infer the object identity from the collected properties and then recall its affordance. For example, "sharp, thin, made of metal" suggests a knife, which affords cutting. We test this with a self-probe: for four Questioners, we sample 200 games that each model itself solved, give the model its own Q&A transcript, and ask it to name the hidden object. Even on games they solved, models fail to name the object in around 90% of them (QWEN3-14B 3.5%, DEEPSEEK-V4-PRO 5.5%, GPT-5 9.2%, GEMINI-2.5-PRO 8.5%). Although each answer reveals one physical property, identifying the affordance requires far fewer properties than identifying the object, so the game usually ends before enough information accumulates to reveal the identity.

Robustness to counterintuitive properties. We further modify the object properties in 30 games (e.g., a bladed tool's material is rewritten as soft and flexible) so that one original distractor becomes the new target, while the original target remains among the candidates as a tempting distractor. We run the same 20Q game on these modified games and compare with their original versions (Table 2). The models perform comparably on the modified games and rarely pick the tempting distractor, showing that our game structure ensures models collect physical properties through questioning and reasoning over them, rather than recalling the affordance of a typical object. We note that the number of feasible games is small, since AFFORDANCE20Q was not designed for this purpose, and a more comprehensive study requires collecting suitable data, which we leave to future work.

	SR (%)		Tempted (%)
Questioner	Original	Modified	Modified
QWEN3-14B	20.0	50.0	0.0
DEEPSEEK-V4-PRO	36.7	30.0	16.7
GPT-5	43.3	30.0	6.7
GEMINI-2.5-PRO	43.3	50.0	0.0

Table 2: Results on the 30 games with counterintuitive properties. SR is the success rate on the original and the modified games. Tempted is the share of modified games in which the model picks the tempting distractor as its final guess.

8 Conclusion

We introduced AFFORDANCE20Q, a 20-Questions benchmark that measures affordance reasoning from physical properties rather than object-identity recall, comprising 1,009 games over 454 objects and 59 affordances. Our experiments with 15 state-of-the-art LLMs reveal a substantial gap (∼20 points) compared to human performance, and a KL-based information-gain analysis shows that models fail to ask discriminating questions as the game progresses. To close the gap, we proposed KARI, a pipeline based on LLMs that generates affordance rules grounded in evidence from knowledge bases. KARI improves open-source LLMs by up to 15.2 points, while the limited coverage of KBs hinders further gains. We release the benchmark and all code, and hope AFFORDANCE20Q drives progress on physical reasoning that generalizes beyond memorized object-affordance mappings.

9 Limitations

Scope of physical affordances. AFFORDANCE20Q covers 59 affordances and 454 objects sampled from CSKG. We filter out affordances not deducible from physical-property dimensions alone, since their mechanisms cannot be reached by any material or shape question. For example, an induction cooktop heats via internal circuitry. Although some benchmarks (Wang et al., 2026b) make initial attempts in this direction, a systematic study from non-physical properties to such affordances remains unexplored.

Text-only domain. AFFORDANCE20Q is the first work to test a model's affordance reasoning ability from physical properties without exposing the object's identity. However, the whole benchmark is set up in a text-only setting. Exploring a similar setting in the visual domain can provide more insights into how current MLLMs perform on affordance reasoning, which also aligns with many real-world scenario tasks (Jiang et al., 2026; Driess et al., 2023; Zhang et al., 2025; Jiang et al., 2024).

Question-Answering Setting. Our benchmark formulates affordance reasoning in a question-answering (QA) setting. Real-world applications often require models to reason about affordances embedded in richer contexts, such as narrative understanding (Jiang et al., 2023a; Kočiskỳ et al., 2018), open-ended planning (Wang et al., 2023a) or long-horizon memories (Liu et al., 2026b), where affordance cues are implicit and must be inferred from context rather than elicited through explicit queries. Extending AFFORDANCE20Q to these settings can be a promising direction for future work.

Outcome-based evaluation. Our main metric, success rate, only checks the final guess, while the 20Q game naturally has a step-wise structure. Prior work has shown that final accuracy alone can be unreliable for evaluating reasoning (Sun et al., 2026; Fan et al., 2026; Sheng et al., 2026), since a model can reach the correct answer through inconsistent or hallucinated intermediate steps. Our information-gain analysis is a first step toward evaluating the process, and future work can build process-level metrics on the game transcripts to evaluate the hallucination and consistency of models' reasoning.

10 Ethical Considerations

Data sources. Our affordance vocabulary and positive evidence are drawn from CSKG (Ilievski et al., 2021), which is publicly released for research use. The benchmark contains only physical-property descriptions of everyday objects and includes no personal or sensitive data.

Intended use and potential risks. AFFORDANCE20Q is intended for research on physical affordance reasoning. While prior work has shown can be exploited to elicit harmful behaviors from LLMs through multi-turn attacks (Jiang et al., 2025; Wang et al., 2025), our task is strictly confined to querying physical properties of everyday objects, and thus the benchmark and KARI rules contain no offensive content. We note that closed-source LLM scores reported here may shift as providers update their underlying models.

Acknowledgments

This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Agreement No. HR00112590089. This research was sponsored by the Defense Advanced Research Projects Agency via Contract HR00112390061.

References

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, and 1 others. 2024. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

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*, pages 342–364.

Stéphane Aroca-Ouellette, Cory Paik, Alessandro Roncone, and Katharina Kann. 2021. PROST: Physical reasoning about objects through space and time. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4597–4608, Online. Association for Computational Linguistics.

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*, pages 153–162.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, and 1 others. 2020. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439.

Aaron Blakeman, Aaron Grattafiori, Aarti Basant, Abhibha Gupta, Abhinav Khattar, Adi Renduchintala, Aditya Vavre, Akanksha Shukla, Akhiad Bercovich, Aleksander Ficek, and 1 others. 2025. Nvidia nemotron 3: Efficient and open intelligence. *arXiv preprint arXiv:2512.20856*.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 4762–4779.

Jerome Seymour Bruner, Rose R Olver, Patricia M Greenfield, and 1 others. 1966. Studies in cognitive growth.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Bhavana Dalvi, Niket Tandon, and Peter Clark. 2017. Domain-targeted, high precision knowledge extraction. *Transactions of the Association for Computational Linguistics*, 5:233–246.

DeepSeek-AI. 2026. DeepSeek-V4.

Shengheng Deng, Xun Xu, Chaozheng Wu, Ke Chen, and Kui Jia. 2021. 3d affordancenet: A benchmark for visual object affordance understanding. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1778–1787.

Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, and 1 others. 2023. Palm-e: an embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, pages 8469–8488.

Karl Duncker and Lynne S Lees. 1945. On problem-solving. *Psychological monographs*, 58(5):i.

Zhichao Fan, Yanhang Li, and Zexin Zhuang. 2026. Chains that see, answers that don't: A multi-aspect evaluation recipe for forced chain-of-thought on video-mme. *arXiv preprint arXiv:2606.22862*.

Christiane Fellbaum. 2010. Wordnet. In *Theory and applications of ontology: computer applications*, pages 231–243. Springer.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Tim P German and Margaret Anne Defeyter. 2000. Immunity to functional fixedness in young children. *Psychonomic Bulletin & Review*, 7(4):707–712.

James J Gibson. 1977. The theory of affordances. *Hilldale, USA*, 1(2):67–82.

Magnus F Gjerde, Vanessa Cheung, and David Lagnado. 2025. Reasoning about affordances: Causal and compositional reasoning in llms. *arXiv preprint arXiv:2502.16606*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Dylan Hutson, Daniel Vennemeyer, Aneesh Deshmukh, Justin Zhan, and Tianyu Jiang. 2025. Guessinggame: Measuring the informativeness of open-ended questions in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 17344–17360.

Jena D Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. 2021. (comet-) atomic 2020: On symbolic and neural commonsense knowledge graphs. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 6384–6392.

Filip Ilievski, Pedro Szekely, and Bin Zhang. 2021. Cskg: The commonsense knowledge graph. In *European Semantic Web Conference*, pages 680–696. Springer.

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 (Volume 1: Long Papers)*, pages 6941–6951.

Yifan Jiang, Kriti Aggarwal, Tanmay Laud, Kashif Munir, Jay Pujara, and Subhabrata Mukherjee. 2025. Red queen: Exposing latent multi-turn risks in large language models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 25554–25591.

Yifan Jiang, Filip Ilievski, and Kaixin Ma. 2023a. Transferring procedural knowledge across commonsense tasks. In *26th European Conference on Artificial Intelligence, ECAI 2023*, pages 1156–1163. IOS Press BV.

Yifan Jiang, Filip Ilievski, Kaixin Ma, and Zhivar Sourati. 2023b. Brainteaser: Lateral thinking puzzles for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14317–14332.

Yifan Jiang, Yueying Wang, Rui Zhao, Toufiq Parag, Zhimin Chen, Zhenyu Liao, and Jayakrishnan Unnikrishnan. 2026. Videop2r: Video understanding from perception to reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8303–8313.

Yifan Jiang, Jiarui Zhang, Kexuan Sun, Zhivar Sourati, Kian Ahrabian, Kaixin Ma, Filip Ilievski, and Jay Pujara. 2024. Marvel: Multidimensional abstraction and reasoning through visual evaluation and learning. *Advances in Neural Information Processing Systems*, 37:46567–46592.

Katarzyna Kobalczyk, Nicolas Astorga, Tennison Liu, and Mihaela van der Schaar. 2025. Active task disambiguation with llms. *arXiv preprint arXiv:2502.04485*.

Tomáš Kočiskỳ, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Shuyue S Li, Vidhisha Balachandran, Shangbin Feng, Jonathan S Ilgen, Emma Pierson, Pang W Koh, and Yulia Tsvetkov. 2024. Mediq: Question-asking llms and a benchmark for reliable interactive clinical reasoning. *Advances in Neural Information Processing Systems*, 37:28858–28888.

Yong-Lu Li, Yue Xu, Xinyu Xu, Xiaohan Mao, Yuan Yao, Siqi Liu, and Cewu Lu. 2023. Beyond object recognition: A new benchmark towards object concept learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 20029–20040.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, and 1 others. 2022. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*.

Alexander H Liu, Kartik Khandelwal, Sandeep Subramanian, Victor Jouault, Abhinav Rastogi, Adrien Sadé, Alan Jeffares, Albert Jiang, Alexandre Cahill, Alexandre Gavaudan, and 1 others. 2026a. Ministral 3. *arXiv preprint arXiv:2601.08584*.

Bingchen Liu, Yuanyuan Fang, Lei Liu, Guangyuan Dong, Xing Fu, Yuanyuan Gao, Shuyue Wei, Xin Li, and Xiangtian Meng. 2026b. Conmem: Contribution-aware memory for long-horizon manufacturing inspection logs. *arXiv preprint arXiv:2607.28126*.

Hongchen Luo, Wei Zhai, Jing Zhang, Yang Cao, and Dacheng Tao. 2022. Learning affordance grounding from exocentric images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2252–2261.

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*, pages 5064–5074, Miami, Florida, USA. Association for Computational Linguistics.

MiniMax. 2025. MiniMax-M2.

Anh Nguyen, Dimitrios Kanoulas, Darwin G Caldwell, and Nikos G Tsagarakis. 2017. Object-based affordances detection with convolutional neural networks and dense conditional random fields. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5908–5915. IEEE.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2022. Sentence-t5: Scalable sentence encoders from pretrained text-to-text models. In *Findings of the association for computational linguistics: ACL 2022*, pages 1864–1874.

Don Norman. 2013. *The design of everyday things: Revised and expanded edition*. Basic books.

Michele Persiani and Thomas Hellström. 2019. Unsupervised inference of object affordance from text corpora. In *Proceedings of the 22nd Nordic Conference on Computational Linguistics*, pages 115–120.

Ehsan Qasemi, Filip Ilievski, Muhao Chen, and Pedro Szekely. 2022. Paco: Preconditions attributed to commonsense knowledge. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6781–6796.

Azzurra Ruggeri, Tania Lombrozo, Thomas L Griffiths, and Fei Xu. 2016. Sources of developmental change in the efficiency of information search. *Developmental psychology*, 52(12):2159.

Jiayi Sheng, Luna Lyu, Jikai Jin, Tanglin Xia, Alex Gu, James Zou, and Pan Lu. 2026. Solving inequality proofs with large language models. *Advances in Neural Information Processing Systems*, 38.

Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, and 1 others. 2025. Openai gpt-5 system card. *arXiv preprint arXiv:2601.03267*.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI conference on artificial intelligence*, volume 31.

Xian Sun, Wei Gao, Yingshuo Wang, Lingdong Kong, Yanhang Li, Zhichao Fan, Zexin Zhuang, Wenlong Dong, Zhiyuan Zheng, Hrishikesh Paranjape, and 1 others. 2026. Beyond accuracy: Measuring

bias acknowledgment in chain-of-thought reasoning for responsible ai evaluation. *arXiv preprint arXiv:2606.15127*.

Yingbo Tang, Shuaike Zhang, Xiaoshuai Hao, Pengwei Wang, Jianlong Wu, Zhongyuan Wang, and Shanghang Zhang. 2025. Affordgrasp: In-context affordance reasoning for open-vocabulary task-oriented grasping in clutter. In *2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9433–9439. IEEE.

Gemma Team and 1 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Yufei Tian, Abhilasha Ravichander, Lianhui Qin, Ronan Le Bras, Raja Marjieh, Nanyun Peng, Yejin Choi, Thomas L Griffiths, and Faeze Brahman. 2024. Macgyver: Are large language models creative problem solvers? In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5303–5324.

Zifu Wan, Yaqi Xie, Ce Zhang, Zhiqiu Lin, Zihan Wang, Simon Stepputtis, Deva Ramanan, and Katia P Sycara. 2025. Instructpart: Task-oriented part segmentation with instruction reasoning. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 24202–24227.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2023a. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*.

Hanqing Wang, Shaoyang Wang, Yiming Zhong, Zemin Yang, Jiamin Wang, Zhiqing Cui, Jiahao Yuan, Yifan Han, Mingyu Liu, and Yuexin Ma. 2026a. Affordance-r1: Reinforcement learning for generalizable affordance reasoning in multimodal large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 9738–9746.

Junying Wang, Wenzhe Li, Yalun Wu, Yingji Liang, Yijin Guo, Chunyi Li, Haodong Duan, Zicheng Zhang, and Guangtao Zhai. 2026b. Affordance benchmark for mllms. In *ICASSP 2026-2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 13237–13241. IEEE.

Kun Wang, Guibin Zhang, Zhenhong Zhou, Jiahao Wu, Miao Yu, Shiqian Zhao, Chenlong Yin, Jinhu Fu, Yibo Yan, Hanjun Luo, and 1 others. 2025. A comprehensive survey in llm (-agent) full stack safety: Data, training and deployment. *arXiv preprint arXiv:2504.15585*.

Yi Wang, Jiafei Duan, Dieter Fox, and Siddhartha Srinivasa. 2023b. Newton: Are large language models capable of physical reasoning? In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9743–9758.

Peter West, Chandra Bhagavatula, Jack Hessel, Jena Hwang, Liwei Jiang, Ronan Le Bras, Ximing Lu, Sean Welleck, and Yejin Choi. 2022. Symbolic knowledge distillation: from general language models to commonsense models. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4602–4625.

Zhaofeng Wu, Linlu Qiu, Alexis Ross, Ekin Akyürek, Boyuan Chen, Bailin Wang, Najoung Kim, Jacob Andreas, and Yoon Kim. 2024. Reasoning or reciting? exploring the capabilities and limitations of language models through counterfactual tasks. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1819–1862.

Chao Xu, Yixin Chen, He Wang, Song-Chun Zhu, Yixin Zhu, and Siyuan Huang. 2022. Partafford: Part-level affordance discovery from 3d objects. *arXiv preprint arXiv:2202.13519*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Chunlin Yu, Hanqing Wang, Ye Shi, Haoyang Luo, Sibei Yang, Jingyi Yu, and Jingya Wang. 2025. Seqafford: Sequential 3d affordance reasoning via multimodal large language model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1691–1701.

Guibin Zhang, Hejia Geng, Xiaohang Yu, Zhenfei Yin, Zaibin Zhang, Zelin Tan, Heng Zhou, Zhongzhi Li, Xiangyuan Xue, Yijiang Li, and 1 others. 2025. The landscape of agentic reinforcement learning for llms: A survey. *arXiv preprint arXiv:2509.02547*.

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 (Volume 1: Long Papers)*, pages 1495–1516.

Xiaomeng Zhu, Yuyang Li, Leiyao Cui, Pengfei Li, Huan-ang Gao, Yixin Zhu, and Hao Zhao. 2025. Afford-x: Generalizable and slim affordance reasoning for task-oriented manipulation. *arXiv preprint arXiv:2503.03556*.

Yuke Zhu, Alireza Fathi, and Li Fei-Fei. 2014. Reasoning about object affordances in a knowledge base representation. In *European conference on computer vision*, pages 408–424. Springer.

A KARI Pipeline

A.1 Prompts

We list the system prompts for three components as follows:

Proposer. The proposer writes the rule given the affordance verbs and corresponding positive objects with physical properties from KB.

```
You are a physical reasoning expert. Given an
    affordance verb and
example objects that can perform it, produce a
    rule describing the
PHYSICAL PREREQUISITES.

## GRAMMAR (output valid JSON only)

Logic (2) -- combine sub-expressions:
  {"op": "AND", "args": [expr, ...]}
  {"op": "OR", "args": [expr, ...]}

Constructor (1) -- introduce a named part of the
     object:
  {"op": "PART", "role": "<var>", "body": expr}

Atomic predicates (4) -- attach a static
    physical property:
  {"op": "MADE_OF", "role": "<whole|var>", "
      class": "<material>"}
  {"op": "SHAPE", "role": "<whole|var>", "class":
       "<shape>"}
  {"op": "SIZE", "role": "<whole|var>", "class":
       "<size>"}
  {"op": "SURFACE", "role": "<whole|var>", "
      class": "<surface>"}

## ROLE SEMANTICS
- "whole" = the entire object.
- "x" / "y" / "z" = named parts introduced by
    PART.

## CLASS VALUES
Use concise canonical terms describing STATIC
    PHYSICAL structure
(composition, geometry, size, surface texture).

## HARD RULES
- NEVER use affordance verbs or their adjective
    forms
  (forbidden: "cuttable", "reflective", "
      absorbent").
- NEVER use dynamic behavior words
  (forbidden: "compressible", "foldable");
      describe the static
  property that enables the behavior instead (e.
      g. "foam" not
  "compressible").
- Every PART must introduce a fresh variable
    name.
- Every atom's "role" must be "whole" or a
    previously declared
  PART variable (no free variables).

## OUTPUT
JSON rule only. No markdown, no prose.
```

Validator. The validator labels each atom of the current rule *YES*/*NO* for a given object.

```
For each (op, class) atom, decide YES/NO: does
    the given object have
this STATIC PHYSICAL property, based on the raw
    evidence plus your
world knowledge of the object?

A PART-level atom (e.g. SHAPE=sharp_edge) holds
    if ANY part of the
object has the property -- not only the whole. E.
    g. a knife has
SHAPE=sharp_edge because its blade does.

## OUTPUT (JSON array)
[{"op": "<OP>", "class": "<val>", "holds": true|
    false}, ...]

## OUTPUT FORMAT
JSON array only. No markdown, no prose.
```

Auditor. The auditor checks the value of each predicate and updates new ones into the vocabulary.

```
For each proposed class value (none of them
    literally equals any
vocab word -- that's been pre-filtered), decide:
- SYNONYM: near-exact semantic equivalence with
    a vocab word
          -> map to that word.
- NEW: genuinely new concept, no clear synonym
    in vocab
          -> mapped_to = null.

## OP: {op_name}
## VOCABULARY: {vocab_for_this_op}

## OUTPUT (JSON array)
[{"proposed": "<val>", "decision": "SYNONYM|NEW",

  "mapped_to": "<vocab_word>|null"}, ...]

## HARD RULE
SYNONYM requires the SAME underlying concept
    within this op's
semantic dimension. When unsure, output NEW.

## OUTPUT FORMAT
JSON array only. No markdown, no prose.
```

A.2 Rule Examples

We present two rules to illustrate KARI's tree structure. *pierce_through* (Listing 1) shows three alternative physical pathways, each scoped to a specific part. *fold_flat* (Listing 2) shows four alternatives that all describe the whole object.

Listing 1: Rule for *pierce_through*.

```
OR(
  PART(x, AND(SHAPE: pointed, MADE_OF:
      rigid_material, SIZE: thin)),
  PART(x, AND(SHAPE: tapered, MADE_OF: metal)),
  PART(x, AND(SHAPE: cylindrical, MADE_OF: metal,
       SIZE: long))
```

```
)
```

Listing 2: Rule for *fold_flat*.

```
OR(
  AND(MADE_OF: flexible_material, SHAPE: flat,
      SURFACE: smooth),
  AND(MADE_OF: flexible_material, SHAPE:
      rectangular, SIZE: thin),
  AND(MADE_OF: flexible_material, SHAPE:
      cylindrical),
  AND(MADE_OF: flexible_material, SHAPE: curved,
       SIZE: broad)
)
```

B Further Analysis of KARI's Rule Integration

We further explore what makes rule injection harmful. We first measure how often each model's final guess falls on a candidate that has a rule (Table 3). On average, 50.6% of the candidates in each game carry a rule, so a random guess lands on a rule-covered candidate with this probability. The vanilla setting stays close to this level (55–57%), showing that models do not prefer rule-covered candidates on their own. With rules injected, the share jumps to 84–87%, showing that rule injection changes the models' choice preference toward rule-covered candidates.

Guess on a rule-covered candidate	Vanilla	+KARI
Open-source LLMs	56.8%	83.7%
Closed-source LLMs	55.3%	87.2%

Table 3: Share of final guesses that fall on a rule-covered candidate.

We then examine games where the target and distractors are both covered by rules, grouped by the number of rule-covered distractors in the candidate list (Table 4). Groups with 0 and 6 covered distractors are omitted due to small sample sizes. The gain shrinks as more distractors also carry rules, but the rules remain helpful, so the drop does not come from conflicts between the rules and the models' internal knowledge. In summary, rule injection shifts the models' choice toward rule-covered candidates, so models can be misled on games they would have solved without rules. This effect is amplified for models with stronger vanilla performance, since they have more correct answers to lose, and eventually leads to their overall drop.

# Rule-covered distractors	Δ SR (%)	
	Open-source	Closed-source
1	+34.9	+31.2
2	+32.5	+27.8
3	+12.9	+6.1
4	+17.0	+11.5
5	+11.8	+7.5

Table 4: Success rate change from KARI's rules on covered targets, by number of rule-covered distractors.

C Experiment Setup Details

C.1 Fix20Q Baseline Question Selection

The Fixed-Q baseline asks 20 yes/no questions in a fixed order, curated from question produced by all evaluated LLM questioners on the test split (Table 5). As Table 5 shows, the questions asked by LLMs are open-vocabulary and follow a long-tail distribution: the 20 most frequent questions account for only 10.2% of all 373,060 yes/no turns. A deterministic lookup oracle can only answer a predefined question set, which would limit the Questioners' questioning ability, so we adopt an LLM Oracle grounded on the property set, as in prior 20Q work (Bertolazzi et al., 2023; Hutson et al., 2025).

C.2 Entropy-Max Baseline

ENTROPY-MAX uses the same representative objects and stopping threshold (0.9) as OPTIMAL, but has no access to the target. It reaches 51.2% in 4.6 turns, outperforming all LLMs in Table 1 but still falling far short of OPTIMAL. Since its questions never aim at any target, its likelihood often converges on a wrong candidate, showing that narrowing the search space alone is not enough for success.

C.3 Prompts

Questioner. The Questioner plays the active role of the game and asks yes/no questions about the hidden object's static physical properties.

```
You are playing a 20 Questions game about object
     AFFORDANCES.

One of the following affordance labels describes
     the hidden object.
Your goal is to identify it by asking yes/no
    questions about the
object's static physical properties.

Candidate affordances:
  1. <affordance_1>
  2. <affordance_2>
```

#	Dimension	Property	Count (%)	Question
1	Material	rigid	28107 (7.5)	Is the object made of a rigid material?
2	Surface	smooth	21613 (5.8)	Is the object's surface smooth?
3	Material	metal	18816 (5.0)	Is the object made of metal?
4	Material	flexible	18248 (4.9)	Is the object made of a flexible material?
5	Shape	hollow	17952 (4.8)	Is the object hollow?
6	Material	fabric	13436 (3.6)	Is the object made of fabric?
7	Shape	cylindrical	11157 (3.0)	Is the object's shape cylindrical?
8	Shape	hollow	10068 (2.7)	Is the object hollow inside?
9	Material	plastic	9442 (2.5)	Is the object made of plastic?
10	Shape	flat	7172 (1.9)	Is the object flat?
11	Material	wood	6375 (1.7)	Is the object made of wood?
12	Material	fibrous	6047 (1.6)	Is the object made of a fibrous material?
13	Shape	elongated	5969 (1.6)	Is the object's shape elongated?
14	Size	thin	5831 (1.6)	Is the object thin?
15	Shape	rectangular	5593 (1.5)	Is the object's shape rectangular?
16	Surface	rough	5500 (1.5)	Is the object's surface rough?
17	Material	glass	5026 (1.3)	Is the object made of glass?
18	Shape	layered	4159 (1.1)	Is the object layered?
19	Shape	bowl-shaped	4123 (1.1)	Is the object's shape bowl-shaped?
20	Shape	solid	4033 (1.1)	Is the object solid?

Table 5: The 20 most frequent physical-property questions across all evaluation games, with their targeted dimension and count (percentage of the 373,060 yes/no turns).

```
  ...

Ask questions along these FOUR physical
    dimensions:
  - MATERIAL -- what it is made of (composition,
      rigidity/flexibility,
            fibrousness)
  - SHAPE -- geometric outline, parts, OR
     topology
            (hollow / solid / enclosed /
                layered)
  - SIZE -- dimensions (how big / long / thick)
  - SURFACE -- outer surface texture

Rules:
- Ask ONE yes/no question per turn, about
    EXACTLY ONE property from
  EXACTLY ONE dimension above.
- Do NOT ask directly which affordance it is.
- Do NOT ask about the object's function, use,
    purpose, activity,
  or category.

BAD questions to AVOID:
  function / use / category:
    "Is it used for cutting?"
    "Does it absorb liquid?"
    "Is it a tool?"
  multiple properties at once (non-atomic):
    "Is it both metal and pointed?"
    "Is it flexible and smooth?"

When you are confident (or after 20 questions),
    output your final
guess using EXACTLY this format on its own line:
  FINAL_GUESS: <affordance label>
  where <affordance label> must be one of the
     candidates above.

Output format each turn:
QUESTION: <your yes/no question>
```

Oracle. The Oracle sees the full description of the hidden object and replies with a single yes/no per question.

```
You are the Oracle in a 20 Questions game about
    physical objects.
You know the hidden object described below. Your
     task is to answer
yes/no questions about the object's physical
    properties -- shape,
material, size, surface, structural parts, and
    other observable
physical characteristics.

Rules:
- Answer ONLY with "Yes" or "No" (one word).
- The description below is provided as
    supplementary context to
  identify which object you are reasoning about.
- Do NOT reveal the object's name.

Hidden object description:
<object_description>
```

Checker. The Checker classifies each question into one or more `DIMENSION:value` pairs and rejects questions that probe function, use, or category. Output is one pair per line, used by the game loop to enforce question atomicity.

```
Classify the question and list EVERY DIMENSION:
    value pair the
question EXPLICITLY probes, one per line. Do NOT
     infer, expand,
paraphrase, or add synonyms -- output only what
    the question
literally mentions, using the question's own
    wording for the value.

Dimensions (with one example each):
```

```
  MATERIAL - composition, rigidity/flexibility,
      fibrousness
            "Is it made of metal?" -> MATERIAL:
                metal
            "Is it rigid?" -> MATERIAL:rigid
  SHAPE - geometric outline, the existence of a
      geometric part,
            OR internal/whole-body topology (
                hollow, solid,
            enclosed, layered, ...).
            "Is it cylindrical?" -> SHAPE:
                cylindrical
            "Does it have a sharp edge?" ->
                SHAPE:sharp_edge
            "Is it hollow?" -> SHAPE:hollow
  SIZE - dimension (how big / long / thick /
      small)
            "Is it pocket-sized?" -> SIZE:
                pocket_sized
  SURFACE - outer surface texture
            "Is it smooth?" -> SURFACE:smooth
  FUNCTION - use / purpose / activity / what the
       object does
            "Is it used for cutting?" ->
                FUNCTION:cutting
  CATEGORY - class / type / what kind of object
      it is
            "Is it a tool?" -> CATEGORY:tool

Comparative SIZE -- preserve the comparison
    direction and reference
object. Do NOT collapse a comparison into a
    canonical size word.
  "Is it larger than a hand?" -> SIZE:
      larger_than_hand
  "Is it smaller than a coin?" -> SIZE:
      smaller_than_coin

Single vs multi mapping:
  "Is it long?" -> SIZE:long
  "Is it long and narrow?" -> SIZE:long
                              SIZE:narrow
  "Is it made of metal and shiny?" -> MATERIAL:
      metal
                                    SURFACE:shiny

Output one DIMENSION:value pair per line. No
    extra text.
```

C.4 Implementation Details

We evaluate 15 LLM questioners. Open-source models (Llama-3.1-8B, Ministral-8B, Qwen3-8B/14B, Qwen3.5-9B, Nemotron-9B, Gemma-3-12B, Phi-4-14B) are served locally with sglang on $8 \times$ NVIDIA RTX A6000 GPUs (48 GB each). Closed-source models (GPT-5, GPT-5-MINI, GEMINI-2.5-FLASH, GEMINI-2.5-PRO, MINIMAX-M2.5) and the two DeepSeek-V4 variants are accessed through commercial APIs (OpenAI, Google AI Studio, OpenRouter). All decoding uses temperature 0, with a per-call `max_tokens` budget of 512 for the Questioner, 20 for the Oracle, and 50 for the Checker. The Oracle and the Checker are both instances of QWEN3-14B served locally by the same sglang backend. Each game runs to a hard cap of 20 turns, after which a forced guess is requested. If the Checker rejects three consecutive questions, the game is terminated and counted as a failure.

Table 6 reports the share of games terminated by the Checker for each Questioner. Most models are affected in fewer than 4% of games. GPT-5-MINI (36.4%) and QWEN3.5-9B (26.4%) are the exceptions, and excluding their terminated games raises their success rates to 35.8% and 34.1%, but the overall ranking and the gap to human performance remain unchanged, so the game format is not the main bottleneck.

Questioner	Terminated (%)	SR	SR excl.
LLAMA-3.1-8B	1.5	18.4	18.7
MINISTRAL-8B	15.8	17.7	21.1
QWEN3-8B	2.2	26.3	26.8
NEMOTRON-9B	13.2	14.9	17.1
QWEN3.5-9B	26.4	25.1	34.1
GEMMA-3-12B	0.0	27.4	27.4
QWEN3-14B	0.2	27.3	27.3
PHI-4-14B	3.8	26.4	27.4
DEEPSEEK-V4-FLASH	1.0	41.3	41.7
DEEPSEEK-V4-PRO	0.9	40.5	40.9
GPT-5-MINI	36.4	22.8	35.8
GPT-5	3.8	35.5	36.9
MINIMAX-M2.5	10.2	34.9	38.9
GEMINI-2.5-FLASH	6.7	41.4	44.4
GEMINI-2.5-PRO	3.4	45.9	47.5

Table 6: Share of games terminated by the Checker, with success rate (%) before and after excluding them.

C.5 Oracle Validation

To verify that the Oracle is accurate enough to drive the game loop, we collect 300 yes/no questions sampled from the test set and have three human annotators label the ground-truth answer for each. We then run three candidate Oracle models on the same set and report agreement with the human labels in Table 8.

We further break down the agreement of QWEN3-14B by property dimension in Table 9. Errors concentrate in SIZE questions that compare the object against an external reference (e.g., "Is the object larger than a shoebox?"), where the answer is not directly stated in the property set.

To test whether the Oracle choice affects the results, we replace QWEN3-14B with GPT-5 as the Oracle and re-run six Questioners on all 1,009 games (Table 7). Success rates shift by at most 2.6 points (mean 1.5), the ranking is unchanged, and the direction of effect of KARI's rules is identical.

	Vanilla		+KARI	
Questioner	QWEN3-14B	GPT-5	QWEN3-14B	GPT-5
QWEN3.5-9B	25.1	23.3	33.2	32.6
QWEN3-14B	27.3	24.7	32.9	34.5
GEMMA-3-12B	27.4	27.9	33.7	34.7
GPT-5-MINI	22.8	24.4	31.2	31.8
GPT-5	35.5	37.9	32.4	34.8
MINIMAX-M2.5	34.9	35.2	35.5	33.7

Table 7: Success rate (%) with QWEN3-14B and GPT-5 as the Oracle.

	QWEN3-8B	QWEN3-14B	GPT-5
Accuracy (%)	85.2	93.4	95.5

Table 8: Oracle agreement with human labels on 300 sampled questions.

QWEN3-14B	MATERIAL	SHAPE	SURFACE	SIZE
Accuracy (%)	98.0	95.7	88.2	84.8

Table 9: Oracle agreement with human labels by property dimension.

Affordance	SR
Top 5	
abrade_surface	57.8
conduct_heat	55.2
enclose_substance	53.3
store_liquid	51.9
transmit_light	51.4
Bottom 5	
hang_from_above	10.6
float_on_water	7.2
hold_between	6.7
ignite	4.9
sink_in_water	3.8

Table 10: Easiest and hardest affordances, averaged over all 15 vanilla questioners.

D Success Rate Breakdown

We report the top-5 easiest and bottom-5 hardest affordances for three model groups: 1) All 15 LLMs (Table 10), 2) Open-source LLMs (Table 11), and 3) Closed-source LLMs (Table 12).

E Human Annotation

E.1 Participants and Instructions.

We recruited annotators for four roles:

(1) Five volunteers who played the human baseline on the 30% subset. The instructions are the same as Questioner (Appendix C.3).

During the games, we observe that participants are limited by the 20-turn budget. Around turn 10, when the collected information is still not decisive, they tend to pick among the 2–3 remaining candidates, and some participants report that they cannot come up with a proper question to further distinguish them.

(2) Three annotators who verified the Oracle answers on a sampled 300-question subset reported in Table 8. The instructions are the same as Oracle (Appendix C.3).

(3,4) Six annotators were involved in our manual annotation and refinement process in Stages 2 and 3 of AFFORDANCE20Qcollection and annotation. Three additional annotators verified our labels for the object-affordance pair. ALL instructions as follows:

```
You are participating in an annotation task to
    construct structured
physical descriptions for everyday objects.

Your goal is to describe each object using only
    observable physical
attributes, including shape, material
    composition, surface
characteristics, dimensions, and structural
    parts.

Guidelines:
- Focus strictly on physical and geometric
    properties.
- Do not include functional, social, or usage-
    related descriptions.
- Materials should be written as material names
    only
  (e.g., metal, wood, plastic, stainless steel).
- Surface and shape attributes should describe
    only physical
  appearance or geometry.
- Part-specific properties should be assigned to
     the corresponding
  object parts rather than the global object
     description.
- The text description should contain 3--6
```

Affordance	SR
Top 5	
abrade_surface	73.3
store_liquid	70.4
transmit_light	65.7
conduct_heat	63.8
enclose_substance	60.9
Bottom 5	
hang_from_above	9.1
float_on_water	8.7
ignite	4.3
hold_between	2.9
sink_in_water	2.6

Table 11: Easiest and hardest affordances for the top-5 open-source models (DEEPSEEK-V4-FLASH, DEEPSEEK-V4-PRO, GEMMA-3-12B, QWEN3-14B, PHI-4-14B).

Affordance	SR
Top 5	
bounce_back	71.4
enclose_substance	70.4
pierce_through	62.6
store_liquid	61.7
abrade_surface	60.0
Bottom 5	
sweep	13.3
hold_between	11.4
ignite	9.6
float_on_water	7.8
sink_in_water	7.0

Table 12: Easiest and hardest affordances for the top-5 closed-source models (GEMINI-2.5-PRO, GEMINI-2.5-FLASH, GPT-5, MINIMAX-M2.5, GPT-5-MINI).

```
    sentences describing
  only physical characteristics.
- Do not include phrases such as "used for", "
    designed to",
  or "can be used".
- Global size should reflect realistic
    dimensions in centimeters.

Please provide the final annotation in a JSON
    format.
```

```
You are participating in an annotation task for
    a physical affordance benchmark.

For each listed object, determine which
    affordances the object
physically supports based only on its observable
     properties,
including shape, geometry, material, surface
    characteristics,
and structural components.

An affordance should be assigned only if the
    object's physical
properties reasonably satisfy the affordance
    definition.

Guidelines:
- Base decisions only on visible or physically
    inferable properties.
- Do not consider electronic, chemical, or
    mechanism-specific
  functions unless they are directly implied by
      the object's
  physical structure.
- If an affordance clearly applies, include it;
    if uncertain,
  leave it unassigned.

Please provide annotations in the following JSON
     format:
[
  {"object": "<name>", "has": ["<affordance_name
      >", ...]},
  ...
]
```

E.2 Recruitment and payment.

All participants were university students who voluntarily participated through internal recruitment channels. Participants did not receive any financial compensation.

E.3 Data Consent

All participants were informed that their annotations would be used solely for academic research and dataset construction purposes prior to participation. And no demographic or personally identifying information was retained beyond the annotation outputs.