Agentic AI

and the Architecture of Trust

About Systems That Act and Humans Who Remain Responsible

Dr. Kirsten MF Weber
Jülich Supercomputing Centre

Dr. Alexander Ebbes
Xyna.AI

OpenRheinMain 2026

AI for Fully Autonomous Weapons?!

Sam Altman,
OpenAI

Pete Hegseth,
Department of War

Dario Amodei,
Anthropic

Source of Picture: https://www.technologyreview.com/2026/03/02/1133850/openais-compromise-with-the-pentagon-is-what-anthropic-feared/

It doesn’t have to be war in some far-away country.
It can start right on your desk.

2022-11-30

The “ChatGPT Moment“ in AI history

1 million users in 5 days,
100 million after 2 months

AI Milestone Timeline

- 2022: Research Preview von ChatGPT
- 2023: Model explosion.
  Everyone talks embeddings, tokens, attention.
  The P in GPT creates an ecosystem.
  But the training corpus is still static.
- 2024: Internet grounding.
  Tools and APIs enable search. Recency, specialization, customization.
- 2025: Agentic AI.
  Action instead of dialogue.
  Writing and executing APIs.
  Expanding reach. Delegated execution. Automation.
- 2026: Trust becomes unavoidable.
  More and more is possible.
  But what do we really want to delegate?

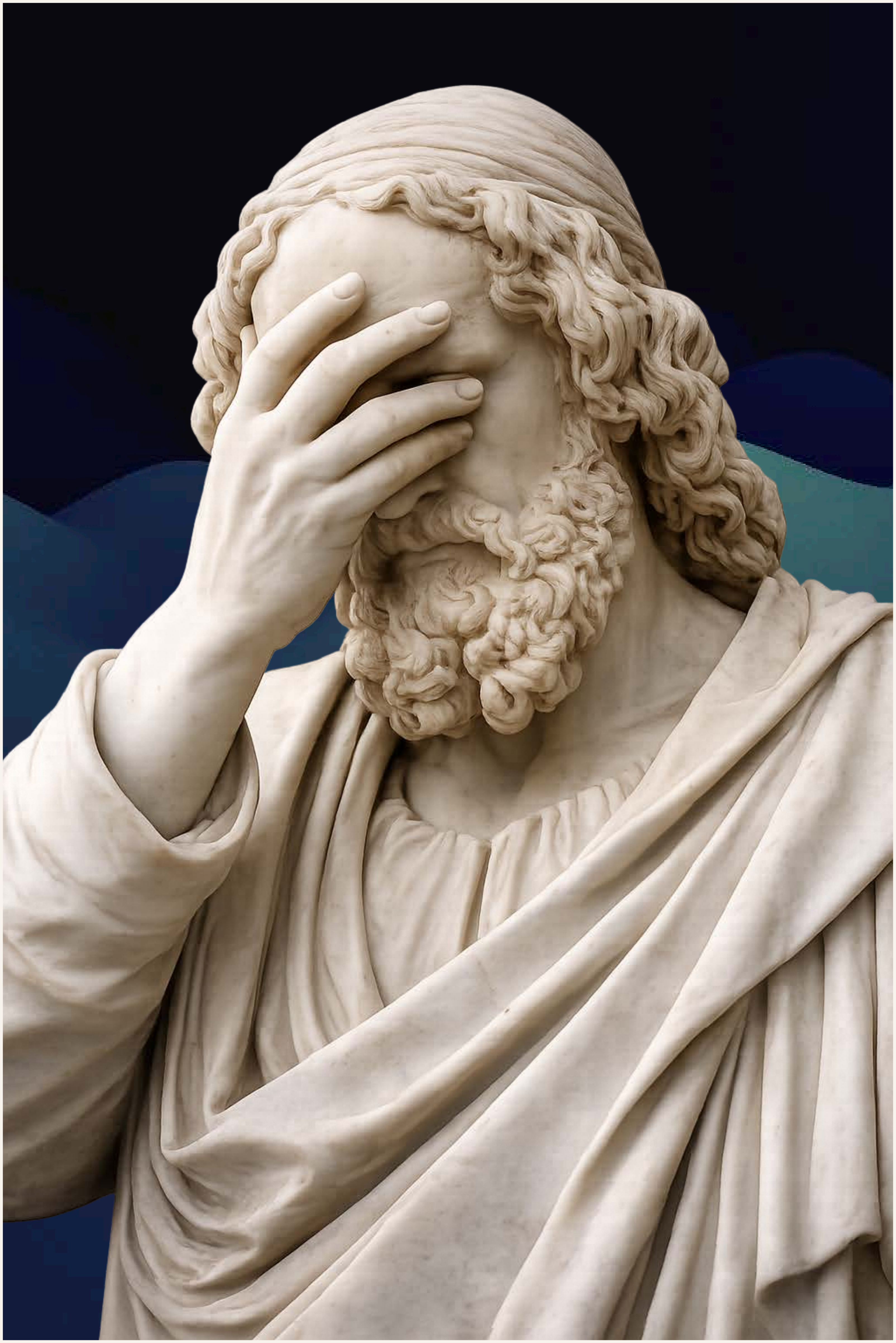
About the Term Trust

- The question of trust is not new.

- Even early users quickly encountered hallucinations.
  AI sometimes just makes things up.

- This is not simply a software bug.
  Hallucination is intrinsic to probabilistic LLMs.
  We understand why it happens —
  but we cannot eliminate it without losing capability.

- The real difference is about impact.
  Misinterpreting a poem is harmless.
  Letting an agent trade your portfolio is not.



Vectors of Harmfulness in AI

- **Hallucination** — AI can fabricate plausible falsehoods
- **Bias** — Asymmetries in training data. Policy-shaped responses.
- **Curation side-effects** —Reducing ambiguity can create content imbalance
- **Data correctness** — Epistemic constraints. Language models approximate knowledge.
- **Reasoning correctness** — What looks like cognition is just high-quality imitation

Psychology: “Fluency-Induced Credibility Bias“ — Confident delivery increases perceived credibility. A foundation of rhetoric, marketing, and propaganda.

“The Illusion of Thinking“

- Controlled **reasoning benchmarks**.
  Tasks like: Tower of Hanoi, River-Crossing puzzles. Complexity can be scaled precisely

- State-of-the-art reasoning models tested.
  ChatGPT O1/O3, Claude Sonnet Thinking, DeepSeek-R1, Gemini Thinking

- **Result**:
  At moderate complexity: models perform well — within distribution.
  Beyond that: reasoning collapses.
  Not gradual degradation. Structural failure

- **Implication**:
  Language models only simulate reasoning.
  They do not perform deduction

arXiv:2506.06941v3 [cs.AI] 20 Nov 2025

The Illusion of Thinking:
Understanding the Strengths and Limitations of Reasoning Models
via the Lens of Problem Complexity

Parshin Shojaee[*†] Iman Mirzadeh[*] Keivan Alizadeh
Maxwell Horton Samy Bengio Mehrdad Farajtabar

Apple

Abstract

Recent generations of frontier language models have introduced Large Reasoning Models (LRMs) that generate detailed thinking processes before providing answers. While these models demonstrate improved performance on reasoning benchmarks, their fundamental capabilities, scaling properties, and limitations remain insufficiently understood. Current evaluations primarily focus on established mathematical and coding benchmarks, emphasizing final answer accuracy. However, this evaluation paradigm often suffers from data contamination and does not provide insights into the reasoning traces' structure and quality. In this work, we systematically investigate these gaps with the help of controllable puzzle environments that allow precise manipulation of compositional complexity while maintaining consistent logical structures. This setup enables the analysis of not only final answers but also the internal reasoning traces, offering insights into how LRMs "think". Through extensive experimentation across diverse puzzles, we show that frontier LRMs face a complete accuracy collapse beyond certain complexities. Moreover, they exhibit a counter-intuitive scaling limit: their reasoning effort increases with problem complexity up to a point, then declines despite having an adequate token budget. By comparing LRMs with their standard LLM counterparts under equivalent inference compute, we identify three performance regimes: (1) low-complexity tasks where standard models surprisingly outperform LRMs, (2) medium-complexity tasks where additional thinking in LRMs demonstrates advantage, and (3) high-complexity tasks where both models experience complete collapse. We found that LRMs have limitations in exact computation: they fail to use explicit algorithms and reason inconsistently across scales and problems. We also investigate the reasoning traces in more depth, studying the patterns of explored solutions and analyzing the models' computational behavior, shedding light on their strengths, limitations, and ultimately raising questions about the nature for their reasoning capabilities.

1 Introduction

Large Language Models (LLMs) have recently evolved to include specialized variants explicitly designed for reasoning tasks—Large Reasoning Models (LRMs) such as OpenAI's o1/o3 [1, 2], DeepSeek-R1 [3], Claude Sonnet Thinking [4], and Gemini Thinking [5]. These models are new artifacts, characterized by their "*thinking*" mechanisms such as long Chain-of-Thought (CoT) with self-reflection, and have demonstrated promising results across various reasoning benchmarks. Their

[*] Equal contribution.
[†] Work done during an internship at Apple.
{pshojaee, imirzadeh, kalizadehvahid, bengio, farajtabar}@apple.com

1

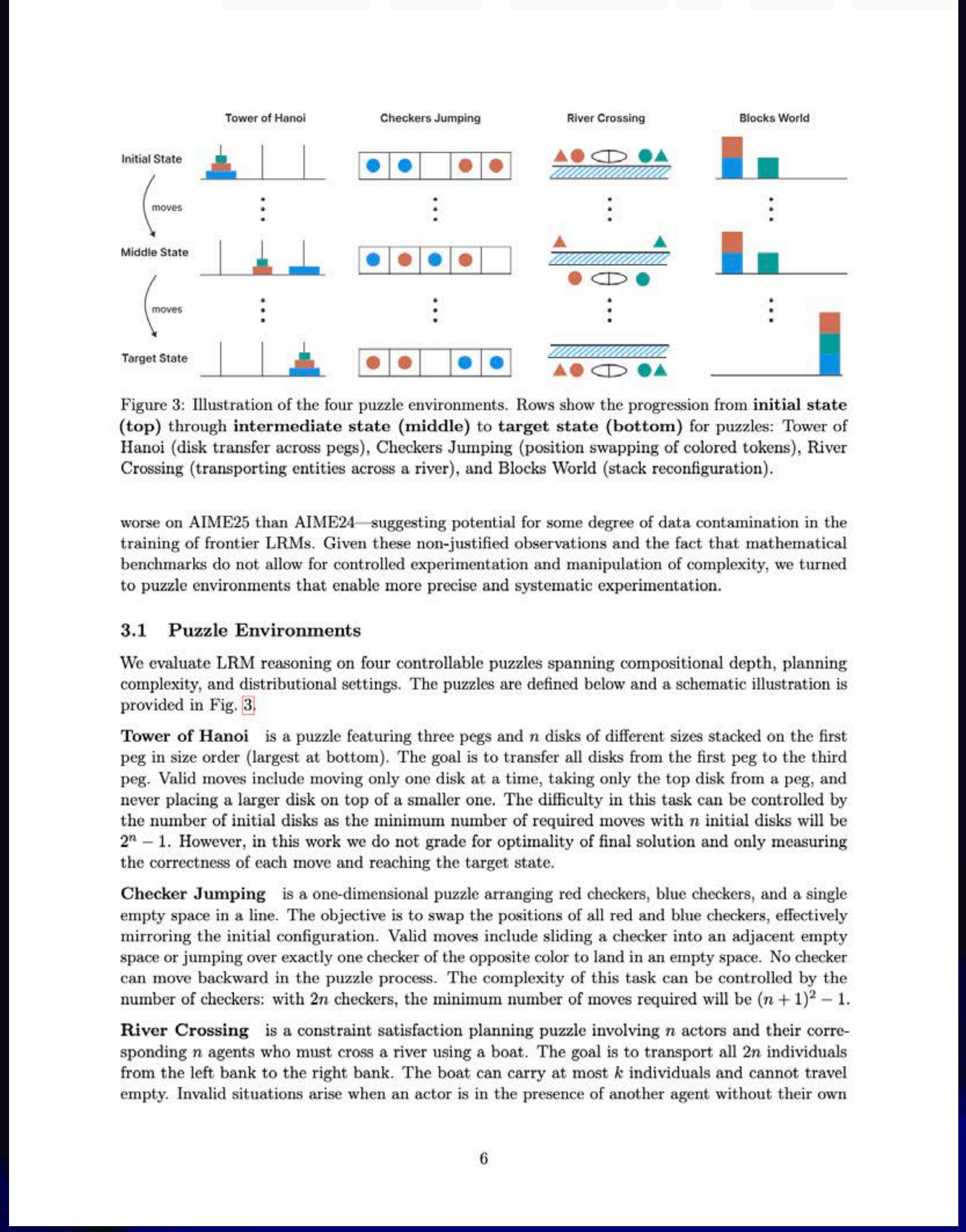


Figure 3: Illustration of the four puzzle environments. Rows show the progression from **initial state (top)** through **intermediate state (middle)** to **target state (bottom)** for puzzles: Tower of Hanoi (disk transfer across pegs), Checkers Jumping (position swapping of colored tokens), River Crossing (transporting entities across a river), and Blocks World (stack reconfiguration).

worse on AIME25 than AIME24—suggesting potential for some degree of data contamination in the training of frontier LRMs. Given these non-justified observations and the fact that mathematical benchmarks do not allow for controlled experimentation and manipulation of complexity, we turned to puzzle environments that enable more precise and systematic experimentation.

3.1 Puzzle Environments

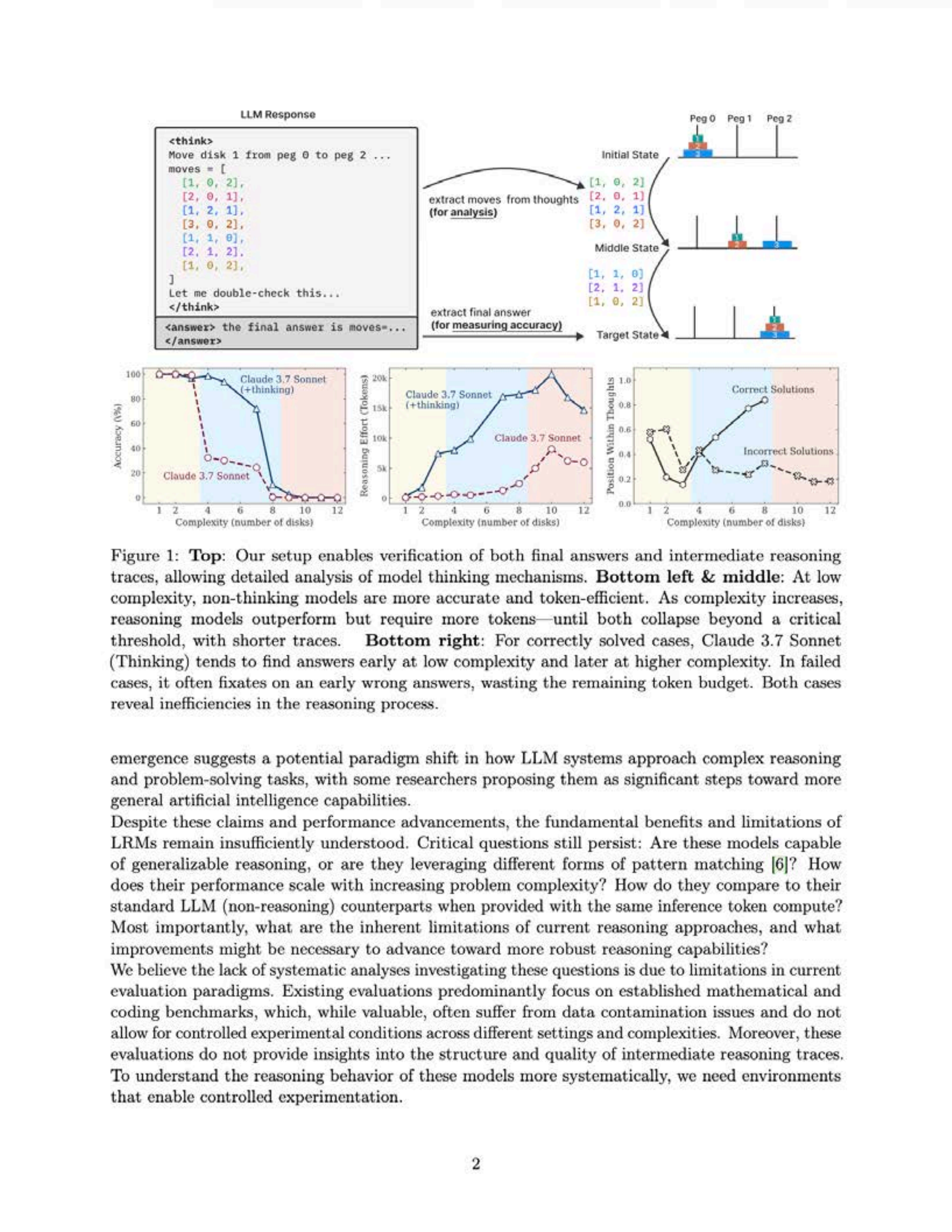
We evaluate LRM reasoning on four controllable puzzles spanning compositional depth, planning complexity, and distributional settings. The puzzles are defined below and a schematic illustration is provided in Fig. 3.

Tower of Hanoi is a puzzle featuring three pegs and n disks of different sizes stacked on the first peg in size order (largest at bottom). The goal is to transfer all disks from the first peg to the third peg. Valid moves include moving only one disk at a time, taking only the top disk from a peg, and never placing a larger disk on top of a smaller one. The difficulty in this task can be controlled by the number of initial disks as the minimum number of required moves with n initial disks will be $2^n - 1$. However, in this work we do not grade for optimality of final solution and only measuring the correctness of each move and reaching the target state.

Checker Jumping is a one-dimensional puzzle arranging red checkers, blue checkers, and a single empty space in a line. The objective is to swap the positions of all red and blue checkers, effectively mirroring the initial configuration. Valid moves include sliding a checker into an adjacent empty space or jumping over exactly one checker of the opposite color to land in an empty space. No checker can move backward in the puzzle process. The complexity of this task can be controlled by the number of checkers: with $2n$ checkers, the minimum number of moves required will be $(n + 1)^2 - 1$.

River Crossing is a constraint satisfaction planning puzzle involving n actors and their corresponding n agents who must cross a river using a boat. The goal is to transport all $2n$ individuals from the left bank to the right bank. The boat can carry at most k individuals and cannot travel empty. Invalid situations arise when an actor is in the presence of another agent without their own

6



LLM Response
extract moves from thoughts (for analysis)
extract final answer (for measuring accuracy)
Initial State
Middle State
Target State
Peg 0
Peg 1
Peg 2
Claude 3.7 Sonnet (+thinking)
Claude 3.7 Sonnet
Correct Solutions
Incorrect Solutions
Complexity (number of disks)

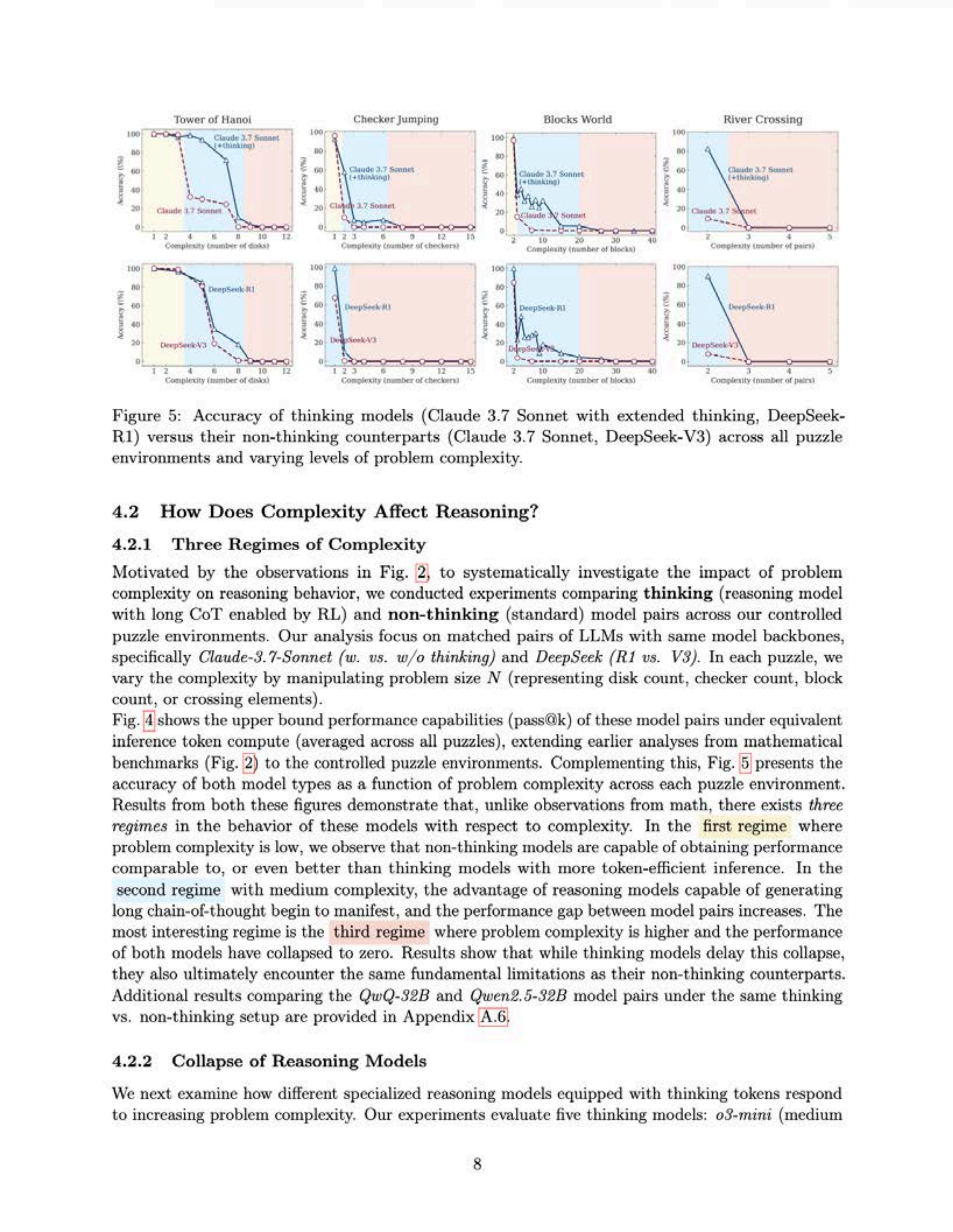
Figure 1: **Top**: Our setup enables verification of both final answers and intermediate reasoning traces, allowing detailed analysis of model thinking mechanisms. **Bottom left & middle**: At low complexity, non-thinking models are more accurate and token-efficient. As complexity increases, reasoning models outperform but require more tokens—until both collapse beyond a critical threshold, with shorter traces. **Bottom right**: For correctly solved cases, Claude 3.7 Sonnet (Thinking) tends to find answers early at low complexity and later at higher complexity. In failed cases, it often fixates on an early wrong answers, wasting the remaining token budget. Both cases reveal inefficiencies in the reasoning process.

emergence suggests a potential paradigm shift in how LLM systems approach complex reasoning and problem-solving tasks, with some researchers proposing them as significant steps toward more general artificial intelligence capabilities.

Despite these claims and performance advancements, the fundamental benefits and limitations of LRMs remain insufficiently understood. Critical questions still persist: Are these models capable of generalizable reasoning, or are they leveraging different forms of pattern matching [6]? How does their performance scale with increasing problem complexity? How do they compare to their standard LLM (non-reasoning) counterparts when provided with the same inference token compute? Most importantly, what are the inherent limitations of current reasoning approaches, and what improvements might be necessary to advance toward more robust reasoning capabilities?

We believe the lack of systematic analyses investigating these questions is due to limitations in current evaluation paradigms. Existing evaluations predominantly focus on established mathematical and coding benchmarks, which, while valuable, often suffer from data contamination issues and do not allow for controlled experimental conditions across different settings and complexities. Moreover, these evaluations do not provide insights into the structure and quality of intermediate reasoning traces. To understand the reasoning behavior of these models more systematically, we need environments that enable controlled experimentation.

2



Tower of Hanoi
Checker Jumping
Blocks World
River Crossing
Claude 3.7 Sonnet (+thinking)
Claude 3.7 Sonnet
DeepSeek-R1
DeepSeek-V3
Complexity (number of disks)
Complexity (number of checkers)
Complexity (number of blocks)
Complexity (number of pairs)

Figure 5: Accuracy of thinking models (Claude 3.7 Sonnet with extended thinking, DeepSeek-R1) versus their non-thinking counterparts (Claude 3.7 Sonnet, DeepSeek-V3) across all puzzle environments and varying levels of problem complexity.

4.2 How Does Complexity Affect Reasoning?

4.2.1 Three Regimes of Complexity

Motivated by the observations in Fig. 2, to systematically investigate the impact of problem complexity on reasoning behavior, we conducted experiments comparing **thinking** (reasoning model with long CoT enabled by RL) and **non-thinking** (standard) model pairs across our controlled puzzle environments. Our analysis focus on matched pairs of LLMs with same model backbones, specifically *Claude-3.7-Sonnet (w. vs. w/o thinking)* and *DeepSeek (R1 vs. V3)*. In each puzzle, we vary the complexity by manipulating problem size N (representing disk count, checker count, block count, or crossing elements).

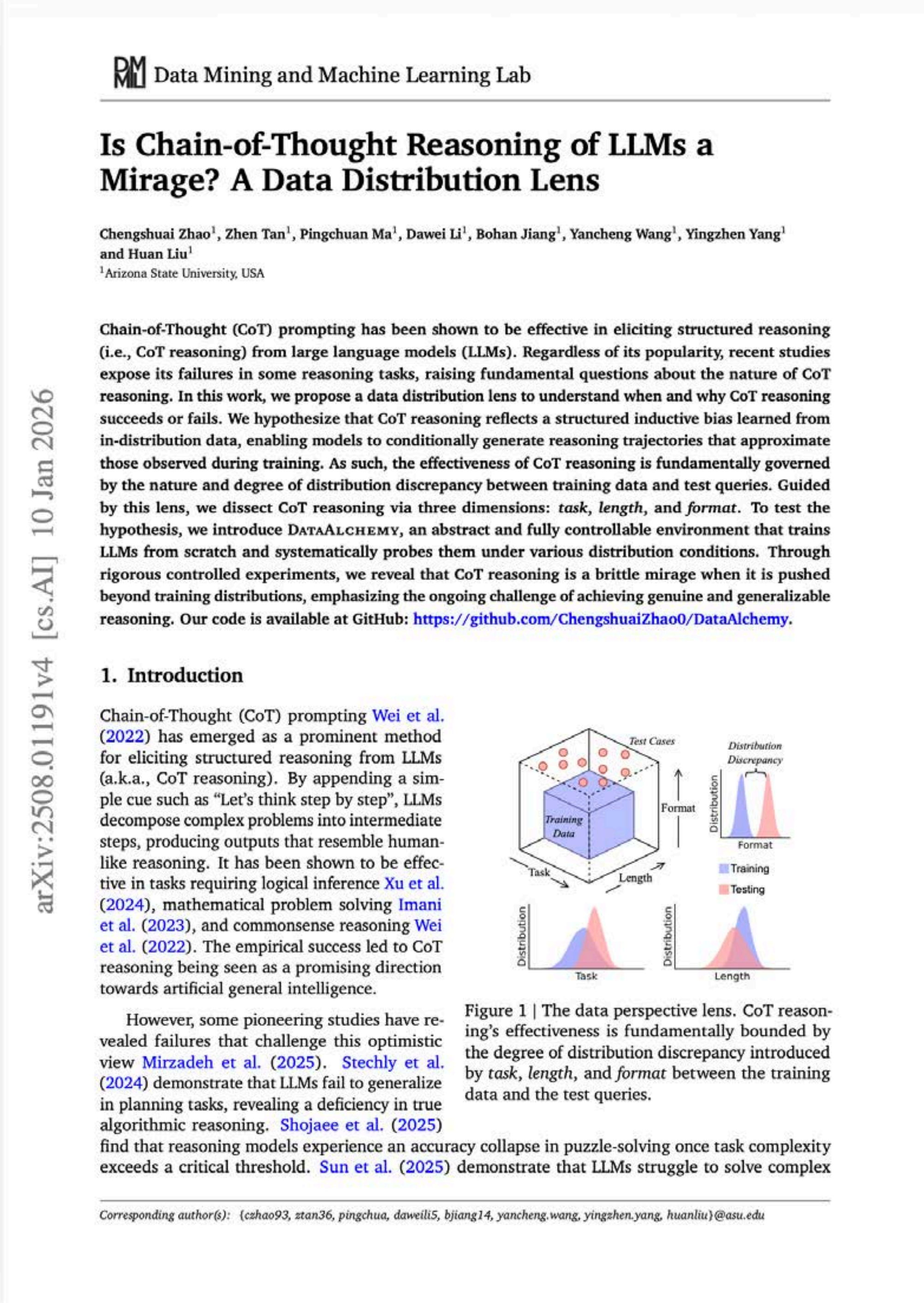
Fig. 4 shows the upper bound performance capabilities (pass@k) of these model pairs under equivalent inference token compute (averaged across all puzzles), extending earlier analyses from mathematical benchmarks (Fig. 2) to the controlled puzzle environments. Complementing this, Fig. 5 presents the accuracy of both model types as a function of problem complexity across each puzzle environment. Results from both these figures demonstrate that, unlike observations from math, there exists *three regimes* in the behavior of these models with respect to complexity. In the first regime where problem complexity is low, we observe that non-thinking models are capable of obtaining performance comparable to, or even better than thinking models with more token-efficient inference. In the second regime with medium complexity, the advantage of reasoning models capable of generating long chain-of-thought begin to manifest, and the performance gap between model pairs increases. The most interesting regime is the third regime where problem complexity is higher and the performance of both models have collapsed to zero. Results show that while thinking models delay this collapse, they also ultimately encounter the same fundamental limitations as their non-thinking counterparts. Additional results comparing the *QwQ-32B* and *Qwen2.5-32B* model pairs under the same thinking vs. non-thinking setup are provided in Appendix A.6.

4.2.2 Collapse of Reasoning Models

We next examine how different specialized reasoning models equipped with thinking tokens respond to increasing problem complexity. Our experiments evaluate five thinking models: *o3-mini* (medium

8

„Is Reasoning a Mirage?“



Data Mining and Machine Learning Lab

Is Chain-of-Thought Reasoning of LLMs a Mirage? A Data Distribution Lens

Chengshuai Zhao[1], Zhen Tan[1], Pingchuan Ma[1], Dawei Li[1], Bohan Jiang[1], Yancheng Wang[1], Yingzhen Yang[1] and Huan Liu[1]

[1]Arizona State University, USA

Chain-of-Thought (CoT) prompting has been shown to be effective in eliciting structured reasoning (i.e., CoT reasoning) from large language models (LLMs). Regardless of its popularity, recent studies expose its failures in some reasoning tasks, raising fundamental questions about the nature of CoT reasoning. In this work, we propose a data distribution lens to understand when and why CoT reasoning succeeds or fails. We hypothesize that CoT reasoning reflects a structured inductive bias learned from in-distribution data, enabling models to conditionally generate reasoning trajectories that approximate those observed during training. As such, the effectiveness of CoT reasoning is fundamentally governed by the nature and degree of distribution discrepancy between training data and test queries. Guided by this lens, we dissect CoT reasoning via three dimensions: *task*, *length*, and *format*. To test the hypothesis, we introduce DataAlchemy, an abstract and fully controllable environment that trains LLMs from scratch and systematically probes them under various distribution conditions. Through rigorous controlled experiments, we reveal that CoT reasoning is a brittle mirage when it is pushed beyond training distributions, emphasizing the ongoing challenge of achieving genuine and generalizable reasoning. Our code is available at GitHub: https://github.com/ChengshuaiZhao0/DataAlchemy.

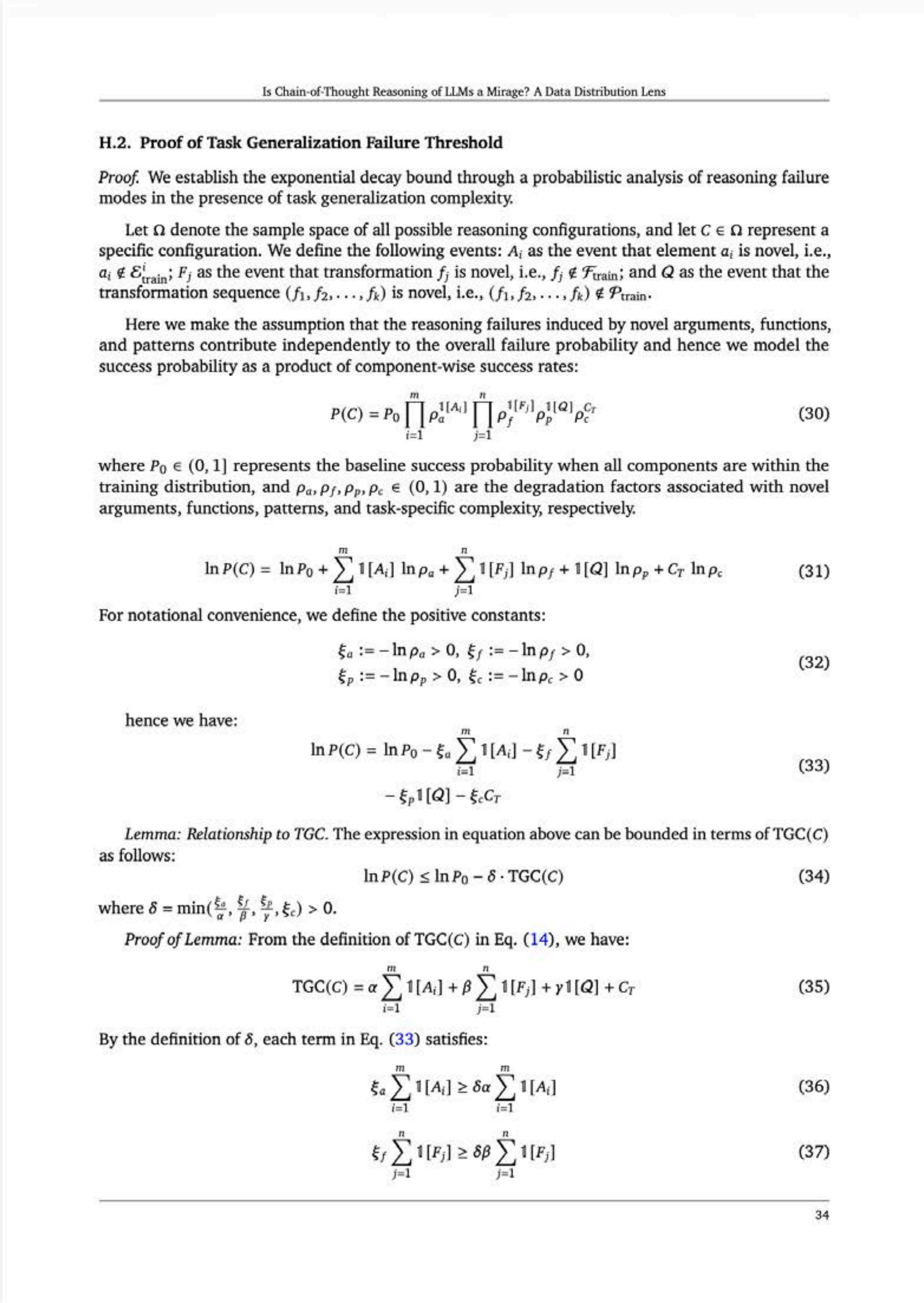
1. Introduction

Chain-of-Thought (CoT) prompting Wei et al. (2022) has emerged as a prominent method for eliciting structured reasoning from LLMs (a.k.a., CoT reasoning). By appending a simple cue such as "Let's think step by step", LLMs decompose complex problems into intermediate steps, producing outputs that resemble human-like reasoning. It has been shown to be effective in tasks requiring logical inference Xu et al. (2024), mathematical problem solving Imani et al. (2023), and commonsense reasoning Wei et al. (2022). The empirical success led to CoT reasoning being seen as a promising direction towards artificial general intelligence.

However, some pioneering studies have revealed failures that challenge this optimistic view Mirzadeh et al. (2025). Stechly et al. (2024) demonstrate that LLMs fail to generalize in planning tasks, revealing a deficiency in true algorithmic reasoning. Shojaee et al. (2025) find that reasoning models experience an accuracy collapse in puzzle-solving once task complexity exceeds a critical threshold. Sun et al. (2025) demonstrate that LLMs struggle to solve complex

Figure 1 | The data perspective lens. CoT reasoning's effectiveness is fundamentally bounded by the degree of distribution discrepancy introduced by *task*, *length*, and *format* between the training data and the test queries.

Corresponding author(s): {czhao93, ztan36, pingchua, daweili5, bjiang14, yancheng.wang, yingzhen.yang, huanliu}@asu.edu



Is Chain-of-Thought Reasoning of LLMs a Mirage? A Data Distribution Lens

H.2. Proof of Task Generalization Failure Threshold

Proof. We establish the exponential decay bound through a probabilistic analysis of reasoning failure modes in the presence of task generalization complexity.

Let Ω denote the sample space of all possible reasoning configurations, and let $C \in \Omega$ represent a specific configuration. We define the following events: A_i as the event that element a_i is novel, i.e., $a_i \notin \mathcal{E}^i_{\text{train}}$; F_j as the event that transformation f_j is novel, i.e., $f_j \notin \mathcal{F}_{\text{train}}$; and Q as the event that the transformation sequence $(f_1, f_2, \dots, f_k)$ is novel, i.e., $(f_1, f_2, \dots, f_k) \notin \mathcal{P}_{\text{train}}$.

Here we make the assumption that the reasoning failures induced by novel arguments, functions, and patterns contribute independently to the overall failure probability and hence we model the success probability as a product of component-wise success rates:

$$P(C) = P_0 \prod_{i=1}^{m} \rho_a^{\mathbb{1}[A_i]} \prod_{j=1}^{n} \rho_f^{\mathbb{1}[F_j]} \rho_p^{\mathbb{1}[Q]} \rho_c^{C_T} \tag{30}$$

where $P_0 \in (0, 1]$ represents the baseline success probability when all components are within the training distribution, and $\rho_a, \rho_f, \rho_p, \rho_c \in (0, 1)$ are the degradation factors associated with novel arguments, functions, patterns, and task-specific complexity, respectively.

$$\ln P(C) = \ln P_0 + \sum_{i=1}^{m} \mathbb{1}[A_i] \ln \rho_a + \sum_{j=1}^{n} \mathbb{1}[F_j] \ln \rho_f + \mathbb{1}[Q] \ln \rho_p + C_T \ln \rho_c \tag{31}$$

For notational convenience, we define the positive constants:

$$\begin{aligned} \xi_a &:= -\ln \rho_a > 0,\ \xi_f := -\ln \rho_f > 0, \\ \xi_p &:= -\ln \rho_p > 0,\ \xi_c := -\ln \rho_c > 0 \end{aligned} \tag{32}$$

hence we have:

$$\begin{aligned} \ln P(C) = \ln P_0 - \xi_a \sum_{i=1}^{m} \mathbb{1}[A_i] - \xi_f \sum_{j=1}^{n} \mathbb{1}[F_j] \\ - \xi_p \mathbb{1}[Q] - \xi_c C_T \end{aligned} \tag{33}$$

Lemma: Relationship to TGC. The expression in equation above can be bounded in terms of TGC(C) as follows:

$$\ln P(C) \le \ln P_0 - \delta \cdot \text{TGC}(C) \tag{34}$$

where $\delta = \min(\frac{\xi_a}{\alpha}, \frac{\xi_f}{\beta}, \frac{\xi_p}{\gamma}, \xi_c) > 0$.

Proof of Lemma: From the definition of TGC(C) in Eq. (14), we have:

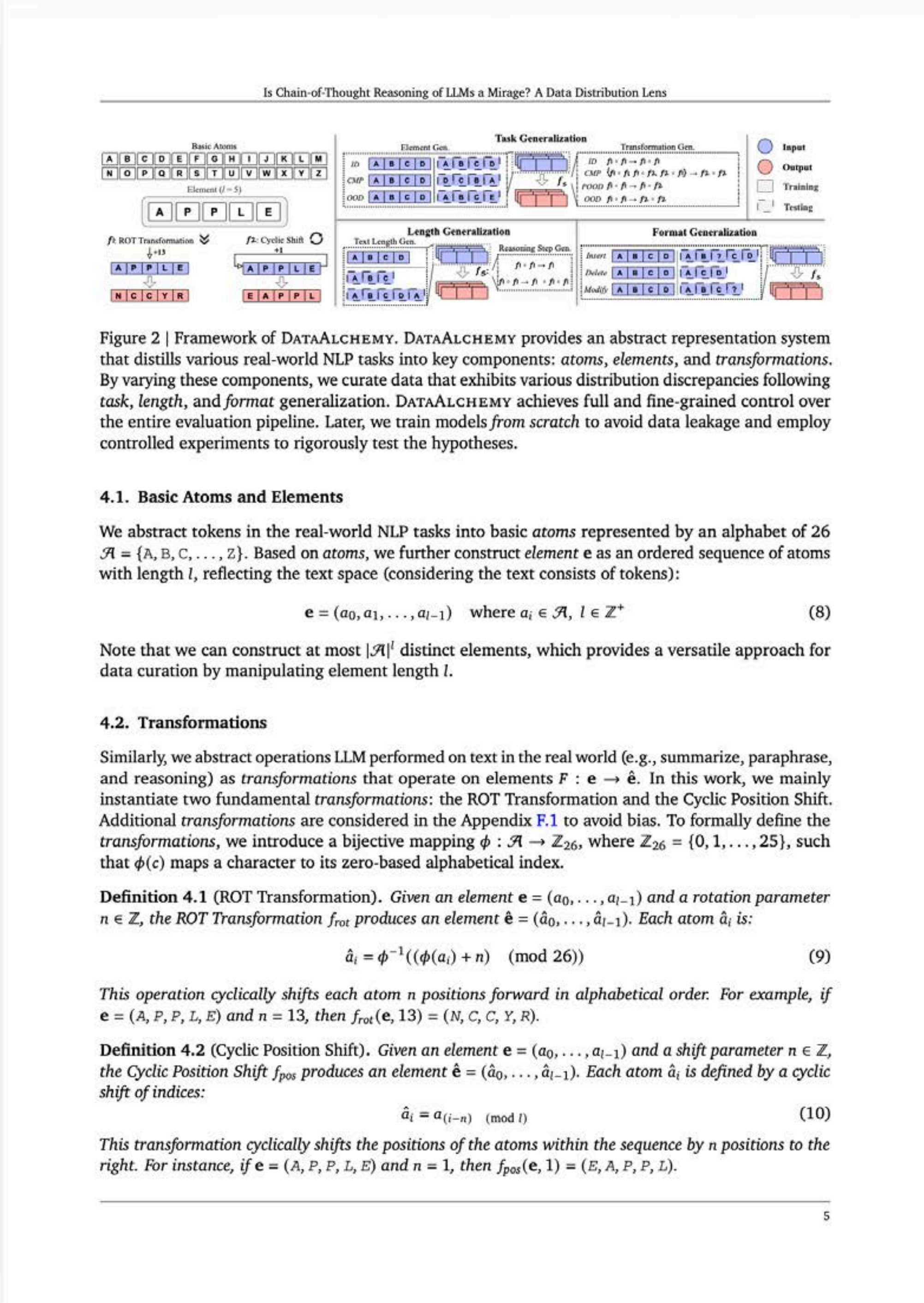
$$\text{TGC}(C) = \alpha \sum_{i=1}^{m} \mathbb{1}[A_i] + \beta \sum_{j=1}^{n} \mathbb{1}[F_j] + \gamma \mathbb{1}[Q] + C_T \tag{35}$$

By the definition of δ, each term in Eq. (33) satisfies:

$$\xi_a \sum_{i=1}^{m} \mathbb{1}[A_i] \ge \delta\alpha \sum_{i=1}^{m} \mathbb{1}[A_i] \tag{36}$$

$$\xi_f \sum_{j=1}^{n} \mathbb{1}[F_j] \ge \delta\beta \sum_{j=1}^{n} \mathbb{1}[F_j] \tag{37}$$

34



Is Chain-of-Thought Reasoning of LLMs a Mirage? A Data Distribution Lens

Figure 2 | Framework of DataAlchemy. DataAlchemy provides an abstract representation system that distills various real-world NLP tasks into key components: *atoms*, *elements*, and *transformations*. By varying these components, we curate data that exhibits various distribution discrepancies following *task*, *length*, and *format* generalization. DataAlchemy achieves full and fine-grained control over the entire evaluation pipeline. Later, we train models *from scratch* to avoid data leakage and employ controlled experiments to rigorously test the hypotheses.

4.1. Basic Atoms and Elements

We abstract tokens in the real-world NLP tasks into basic *atoms* represented by an alphabet of 26 $\mathcal{A} = \{\mathtt{A}, \mathtt{B}, \mathtt{C}, \dots, \mathtt{Z}\}$. Based on *atoms*, we further construct *element* $\mathbf{e}$ as an ordered sequence of atoms with length l, reflecting the text space (considering the text consists of tokens):

$$\mathbf{e} = (a_0, a_1, \dots, a_{l-1}) \quad \text{where } a_i \in \mathcal{A},\ l \in \mathbb{Z}^+ \tag{8}$$

Note that we can construct at most $|\mathcal{A}|^l$ distinct elements, which provides a versatile approach for data curation by manipulating element length l.

4.2. Transformations

Similarly, we abstract operations LLM performed on text in the real world (e.g., summarize, paraphrase, and reasoning) as *transformations* that operate on elements $F : \mathbf{e} \to \hat{\mathbf{e}}$. In this work, we mainly instantiate two fundamental *transformations*: the ROT Transformation and the Cyclic Position Shift. Additional *transformations* are considered in the Appendix F.1 to avoid bias. To formally define the *transformations*, we introduce a bijective mapping $\phi : \mathcal{A} \to \mathbb{Z}_{26}$, where $\mathbb{Z}_{26} = \{0, 1, \dots, 25\}$, such that $\phi(c)$ maps a character to its zero-based alphabetical index.

Definition 4.1 (ROT Transformation). *Given an element $\mathbf{e} = (a_0, \dots, a_{l-1})$ and a rotation parameter $n \in \mathbb{Z}$, the ROT Transformation f_{rot} produces an element $\hat{\mathbf{e}} = (\hat{a}_0, \dots, \hat{a}_{l-1})$. Each atom $\hat{a}_i$ is:*

$$\hat{a}_i = \phi^{-1}((\phi(a_i) + n) \pmod{26}) \tag{9}$$

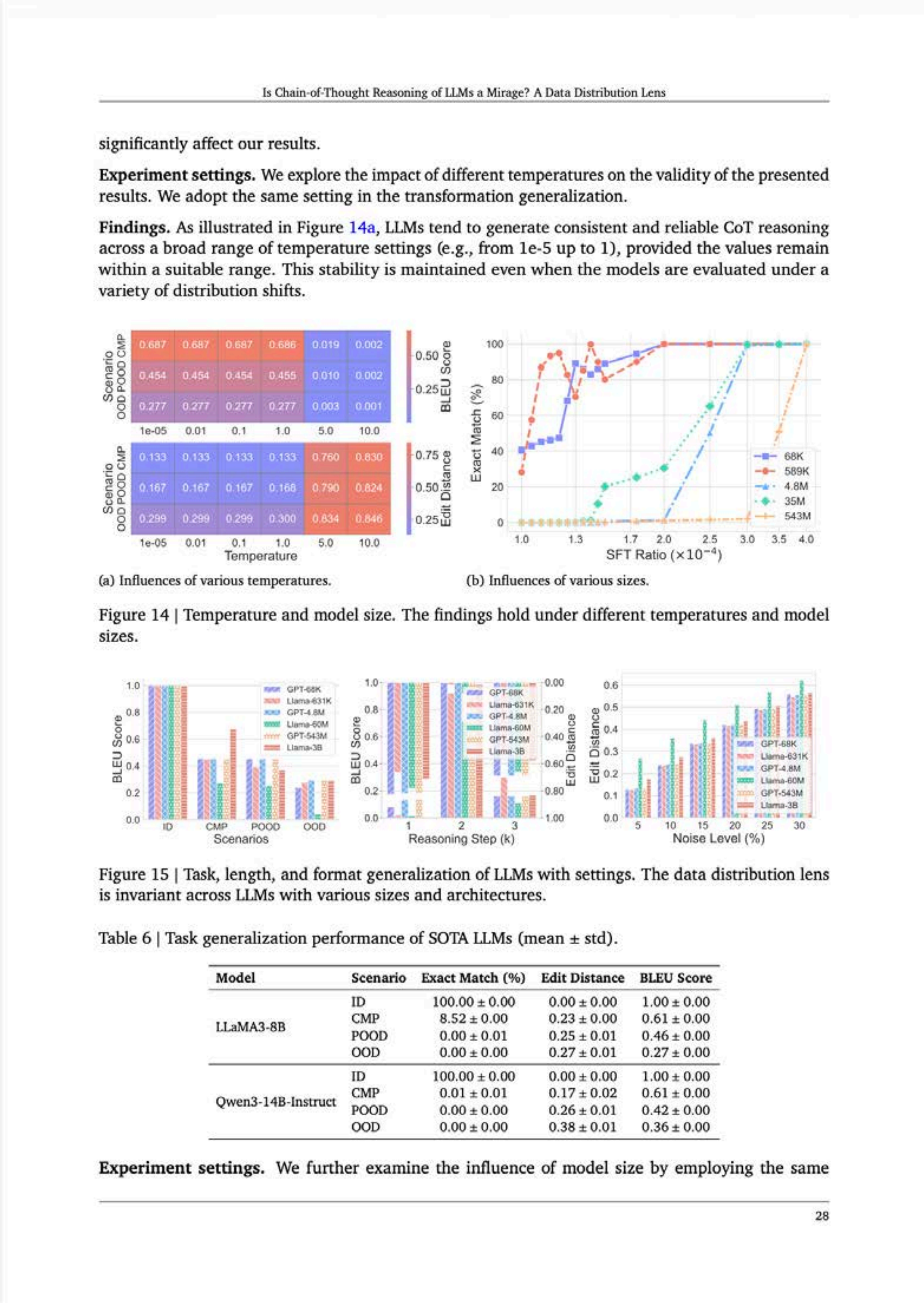
This operation cyclically shifts each atom n positions forward in alphabetical order. For example, if $\mathbf{e} = (A, P, P, L, E)$ and $n = 13$, then $f_{rot}(\mathbf{e}, 13) = (N, C, C, Y, R)$.

Definition 4.2 (Cyclic Position Shift). *Given an element $\mathbf{e} = (a_0, \dots, a_{l-1})$ and a shift parameter $n \in \mathbb{Z}$, the Cyclic Position Shift f_{pos} produces an element $\hat{\mathbf{e}} = (\hat{a}_0, \dots, \hat{a}_{l-1})$. Each atom $\hat{a}_i$ is defined by a cyclic shift of indices:*

$$\hat{a}_i = a_{(i-n) \pmod{l}} \tag{10}$$

This transformation cyclically shifts the positions of the atoms within the sequence by n positions to the right. For instance, if $\mathbf{e} = (A, P, P, L, E)$ and $n = 1$, then $f_{pos}(\mathbf{e}, 1) = (E, A, P, P, L)$.

5



Is Chain-of-Thought Reasoning of LLMs a Mirage? A Data Distribution Lens

significantly affect our results.

Experiment settings. We explore the impact of different temperatures on the validity of the presented results. We adopt the same setting in the transformation generalization.

Findings. As illustrated in Figure 14a, LLMs tend to generate consistent and reliable CoT reasoning across a broad range of temperature settings (e.g., from 1e-5 up to 1), provided the values remain within a suitable range. This stability is maintained even when the models are evaluated under a variety of distribution shifts.

(a) Influences of various temperatures.

(b) Influences of various sizes.

Figure 14 | Temperature and model size. The findings hold under different temperatures and model sizes.

Figure 15 | Task, length, and format generalization of LLMs with settings. The data distribution lens is invariant across LLMs with various sizes and architectures.

Table 6 | Task generalization performance of SOTA LLMs (mean ± std).

Model	Scenario	Exact Match (%)	Edit Distance	BLEU Score
LLaMA3-8B	ID	100.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00
	CMP	8.52 ± 0.00	0.23 ± 0.00	0.61 ± 0.00
	POOD	0.00 ± 0.01	0.25 ± 0.01	0.46 ± 0.00
	OOD	0.00 ± 0.00	0.27 ± 0.01	0.27 ± 0.00
Qwen3-14B-Instruct	ID	100.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00
	CMP	0.01 ± 0.01	0.17 ± 0.02	0.61 ± 0.00
	POOD	0.00 ± 0.00	0.26 ± 0.01	0.42 ± 0.00
	OOD	0.00 ± 0.00	0.38 ± 0.01	0.36 ± 0.00

Experiment settings. We further examine the influence of model size by employing the same

28

- Object of study: Chain-of-Thought reasoning behaviour in LLMs.

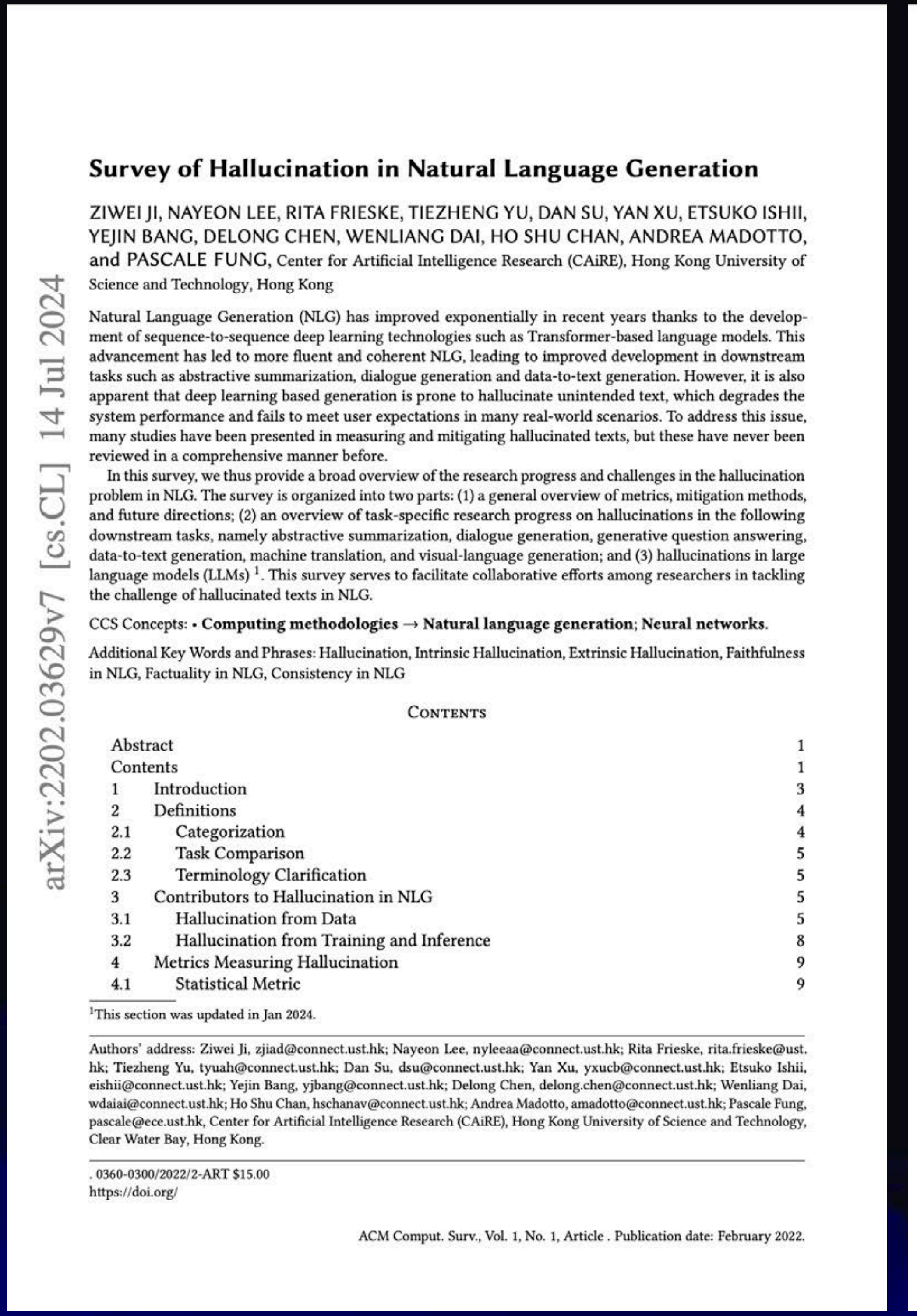
- **Hypothesis**:
  Apparent reasoning is largely determined by the distribution of reasoning trajectories already present in the training corpus.
  Models extend token streams — not logical states.

- **Finding**: Hypothesis confimed.
  "… models conditionally generate reasoning trajectories that approximate those observed during training".

- It is about pattern matching and extrapolation.
  No deduction or thinking. "A brittle mirage".

Hallucination

- Not a software bug. A structural phenomenon.
  - 1986: Already documented in early generative imaging research (e.g. super-resolution from low-resolution inputs). Originally a positive term.
  - Earlier terminology in LLM research → Confabulation. Borrowed from neuropsychology: Patients subconsciously fill memory gaps with invented but subjectively convincing content.
    No intent to deceive.
  - With ChatGPT the catchy term “hallucination“ respawned in public discourse.
  - “Hallucination” implies distorted perception. Machines do not perceive.
    “Confabulation” may be the more precise concept.



arXiv:2202.03629v7 [cs.CL] 14 Jul 2024

Survey of Hallucination in Natural Language Generation

ZIWEI JI, NAYEON LEE, RITA FRIESKE, TIEZHENG YU, DAN SU, YAN XU, ETSUKO ISHII, YEJIN BANG, DELONG CHEN, WENLIANG DAI, HO SHU CHAN, ANDREA MADOTTO, and PASCALE FUNG, Center for Artificial Intelligence Research (CAiRE), Hong Kong University of Science and Technology, Hong Kong

Natural Language Generation (NLG) has improved exponentially in recent years thanks to the development of sequence-to-sequence deep learning technologies such as Transformer-based language models. This advancement has led to more fluent and coherent NLG, leading to improved development in downstream tasks such as abstractive summarization, dialogue generation and data-to-text generation. However, it is also apparent that deep learning based generation is prone to hallucinate unintended text, which degrades the system performance and fails to meet user expectations in many real-world scenarios. To address this issue, many studies have been presented in measuring and mitigating hallucinated texts, but these have never been reviewed in a comprehensive manner before.

In this survey, we thus provide a broad overview of the research progress and challenges in the hallucination problem in NLG. The survey is organized into two parts: (1) a general overview of metrics, mitigation methods, and future directions; (2) an overview of task-specific research progress on hallucinations in the following downstream tasks, namely abstractive summarization, dialogue generation, generative question answering, data-to-text generation, machine translation, and visual-language generation; and (3) hallucinations in large language models (LLMs) [1]. This survey serves to facilitate collaborative efforts among researchers in tackling the challenge of hallucinated texts in NLG.

CCS Concepts: • **Computing methodologies** → **Natural language generation**; **Neural networks**.

Additional Key Words and Phrases: Hallucination, Intrinsic Hallucination, Extrinsic Hallucination, Faithfulness in NLG, Factuality in NLG, Consistency in NLG

Contents

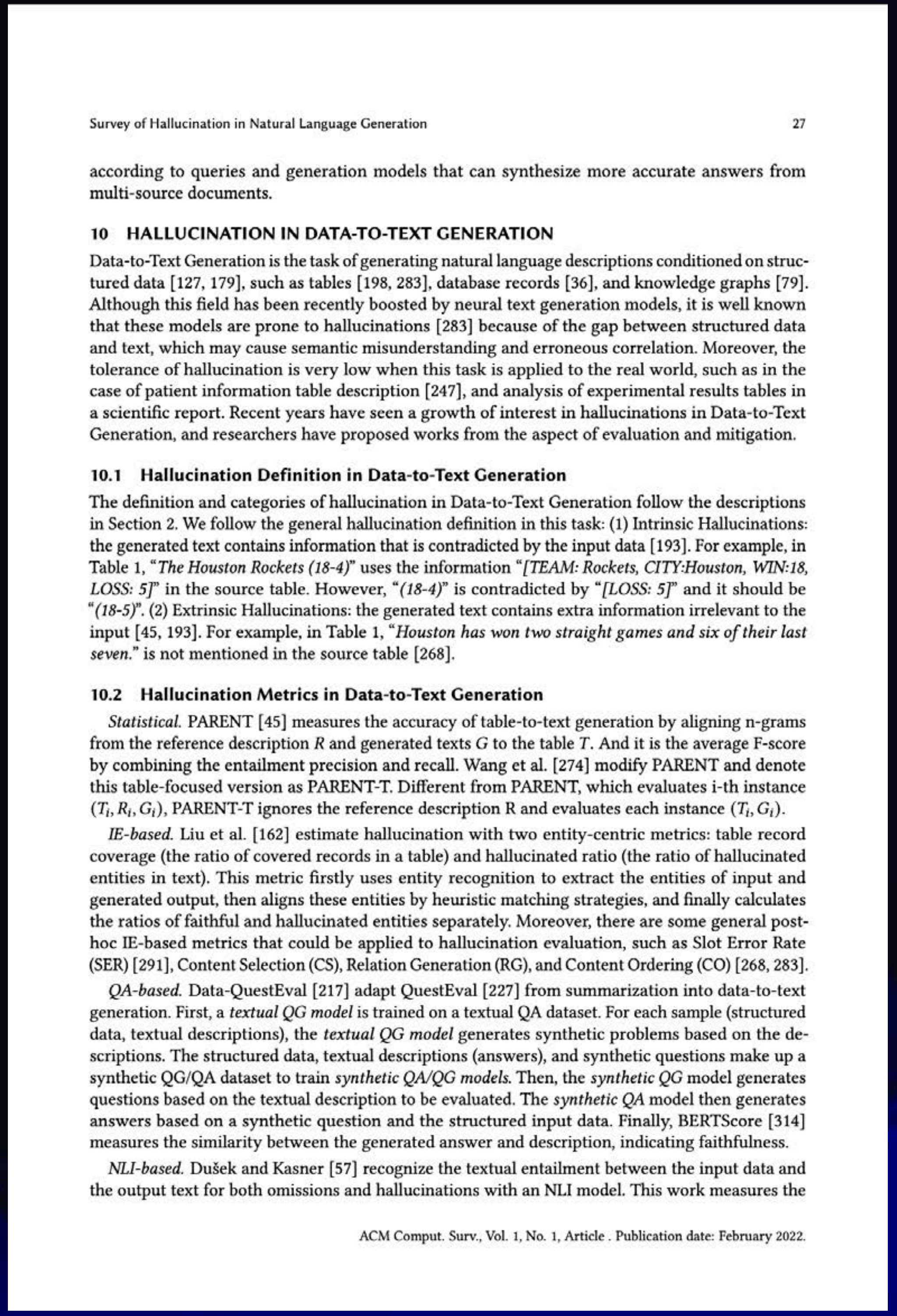
	Abstract	1
	Contents	1
1	Introduction	3
2	Definitions	4
2.1	Categorization	4
2.2	Task Comparison	5
2.3	Terminology Clarification	5
3	Contributors to Hallucination in NLG	5
3.1	Hallucination from Data	5
3.2	Hallucination from Training and Inference	8
4	Metrics Measuring Hallucination	9
4.1	Statistical Metric	9

[1] This section was updated in Jan 2024.

Authors' address: Ziwei Ji, zjiad@connect.ust.hk; Nayeon Lee, nyleeaa@connect.ust.hk; Rita Frieske, rita.frieske@ust.hk; Tiezheng Yu, tyuah@connect.ust.hk; Dan Su, dsu@connect.ust.hk; Yan Xu, yxucb@connect.ust.hk; Etsuko Ishii, eishii@connect.ust.hk; Yejin Bang, yjbang@connect.ust.hk; Delong Chen, delong.chen@connect.ust.hk; Wenliang Dai, wdaiai@connect.ust.hk; Ho Shu Chan, hschanav@connect.ust.hk; Andrea Madotto, amadotto@connect.ust.hk; Pascale Fung, pascale@ece.ust.hk, Center for Artificial Intelligence Research (CAiRE), Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong.

. 0360-0300/2022/2-ART $15.00
https://doi.org/

ACM Comput. Surv., Vol. 1, No. 1, Article . Publication date: February 2022.



Survey of Hallucination in Natural Language Generation 27

according to queries and generation models that can synthesize more accurate answers from multi-source documents.

10 HALLUCINATION IN DATA-TO-TEXT GENERATION

Data-to-Text Generation is the task of generating natural language descriptions conditioned on structured data [127, 179], such as tables [198, 283], database records [36], and knowledge graphs [79]. Although this field has been recently boosted by neural text generation models, it is well known that these models are prone to hallucinations [283] because of the gap between structured data and text, which may cause semantic misunderstanding and erroneous correlation. Moreover, the tolerance of hallucination is very low when this task is applied to the real world, such as in the case of patient information table description [247], and analysis of experimental results tables in a scientific report. Recent years have seen a growth of interest in hallucinations in Data-to-Text Generation, and researchers have proposed works from the aspect of evaluation and mitigation.

10.1 Hallucination Definition in Data-to-Text Generation

The definition and categories of hallucination in Data-to-Text Generation follow the descriptions in Section 2. We follow the general hallucination definition in this task: (1) Intrinsic Hallucinations: the generated text contains information that is contradicted by the input data [193]. For example, in Table 1, "*The Houston Rockets (18-4)*" uses the information "*[TEAM: Rockets, CITY:Houston, WIN:18, LOSS: 5]*" in the source table. However, "*(18-4)*" is contradicted by "*[LOSS: 5]*" and it should be "*(18-5)*". (2) Extrinsic Hallucinations: the generated text contains extra information irrelevant to the input [45, 193]. For example, in Table 1, "*Houston has won two straight games and six of their last seven.*" is not mentioned in the source table [268].

10.2 Hallucination Metrics in Data-to-Text Generation

Statistical. PARENT [45] measures the accuracy of table-to-text generation by aligning n-grams from the reference description R and generated texts G to the table T. And it is the average F-score by combining the entailment precision and recall. Wang et al. [274] modify PARENT and denote this table-focused version as PARENT-T. Different from PARENT, which evaluates i-th instance (T_i, R_i, G_i), PARENT-T ignores the reference description R and evaluates each instance (T_i, G_i).

IE-based. Liu et al. [162] estimate hallucination with two entity-centric metrics: table record coverage (the ratio of covered records in a table) and hallucinated ratio (the ratio of hallucinated entities in text). This metric firstly uses entity recognition to extract the entities of input and generated output, then aligns these entities by heuristic matching strategies, and finally calculates the ratios of faithful and hallucinated entities separately. Moreover, there are some general post-hoc IE-based metrics that could be applied to hallucination evaluation, such as Slot Error Rate (SER) [291], Content Selection (CS), Relation Generation (RG), and Content Ordering (CO) [268, 283].

QA-based. Data-QuestEval [217] adapt QuestEval [227] from summarization into data-to-text generation. First, a *textual QG model* is trained on a textual QA dataset. For each sample (structured data, textual descriptions), the *textual QG model* generates synthetic problems based on the descriptions. The structured data, textual descriptions (answers), and synthetic questions make up a synthetic QG/QA dataset to train *synthetic QA/QG models*. Then, the *synthetic QG* model generates questions based on the textual description to be evaluated. The *synthetic QA* model then generates answers based on a synthetic question and the structured input data. Finally, BERTScore [314] measures the similarity between the generated answer and description, indicating faithfulness.

NLI-based. Dušek and Kasner [57] recognize the textual entailment between the input data and the output text for both omissions and hallucinations with an NLI model. This work measures the

ACM Comput. Surv., Vol. 1, No. 1, Article . Publication date: February 2022.

arXiv:2401.06796v1 [cs.CL] 9 Jan 2024

AI Hallucinations: A Misnomer Worth Clarifying

Negar Maleki
University of South Florida
negarmaleki@usf.edu

Balaji Padmanabhan
University of Maryland
bpadmana@umd.edu

Kaushik Dutta
University of South Florida
duttak@usf.edu

***Abstract*—As large language models continue to advance in Artificial Intelligence (AI), text generation systems have been shown to suffer from a problematic phenomenon termed often as “hallucination.” However, with AI’s increasing presence across various domains including medicine, concerns have arisen regarding the use of the term itself. In this study, we conducted a systematic review to identify papers defining “AI hallucination” across fourteen databases. We present and analyze definitions obtained across all databases, categorize them based on their applications, and extract key points within each category. Our results highlight a lack of consistency in how the term is used, but also help identify several alternative terms in the literature. We discuss implications of these and call for a more unified effort to bring consistency to an important contemporary AI issue that can affect multiple domains significantly[1].**

***Index Terms*—AI, Hallucination, Generative AI**

I. Introduction

One of the early uses of the term "hallucination" in the field of Artificial Intelligence (AI) was in computer vision, in 2000 [1], where it was associated with constructive implications such as super-resolution [1], image inpainting [2], and image synthesis [3]. Interestingly, in this context hallucination was regarded as a valuable asset in computer vision rather than an issue to be circumvented. For instance, an image with low resolution might have been rendered more useful with careful hallucination [1] that generated additional pixels specifically for this purpose.

Despite this (more positive) beginning, recent research has started to employ the term "hallucination" to describe a specific type of error in image captioning [4] and adversarial attack in object detection [5]. In this context, "hallucination" refers to instances where non-existent objects are erroneously detected or incorrectly localized at their anticipated positions. This latter (more negative) interpretation of "hallucination" in computer vision mirrors its analogous usage in language models. For instance, in 2017, researchers highlighted challenges in language models, such as *"the output of the Neural Machine Translation (NMT) system is often quite fluent but entirely unrelated to the input"* [6], or *"language models presume likelihood, but the generated content is ultimately incorrect and unsupported by any information"* [7], which is interpreted as a form of hallucination in AI.

To date, a precise and universally accepted definition of "hallucination" remains absent in the discussions related to this in the increasingly broader field of AI [8]. Diverse definitions, or implied interpretations, persist; sometimes even contradictory, as previously highlighted within the field of computer vision where multiple, disparate interpretations coexist under the same term.

Beyond the AI context, and specifically in the medical domain, the term "hallucination" is a psychological concept denoting a specific form of sensory experience [9]. Ji et al. [10], from the computer science perspective (in ACM Computing Surveys), rationalized the use of the term "hallucination" as *"an unreal perception that feels real"* by drawing from Blom's definition — *"a percept, experienced by a waking individual, in the absence of an appropriate stimulus from the extracorporeal world."* On the other hand, Østergaard et al. [11], from the medical perspective (in Schizophrenia Bulletin, one of the leading journals in the discipline), raised critical concerns regarding even the adoption of the "hallucination" terminology in AI for two primary reasons: 1) The "hallucination" metaphor in AI from this perspective is a misnomer, as AI lacks sensory perceptions, and errors arise from data and prompts rather than the absence of stimuli, and 2) this metaphor is highly stigmatizing, as it associates negative issues in AI with a specific issue in mental illness, particularly schizophrenia, thereby possibly undermining many efforts to reduce stigma in psychiatry and mental health.

Given AI's increasing presence across various domains, including the medical field, concerns have arisen regarding the multifaceted, possibly inappropriate and potentially even harmful use of the term "hallucination" [11], [12]. To address this issue effectively, two potential paths of work offer some promise: 1) The establishment of a consistent and universally applicable terminologies that can be uniformly adopted across all AI-impacted domains will help, particularly if such terminologies lead to the use of more specific and nuanced terms that actually describe the issues they highlight (as we will show later, such vocabulary does exist, but needs more consistent use) and 2) The formulation of a robust and formal definition of "AI hallucination" within the context of AI. These measures are essential to promote clarity and coherence in discussions and research related to "hallucination" in AI, and to mitigate potential confusion and ambiguity in cross-disciplinary applications.

Motivated by these issues, in this paper, we conduct a systematic review of the use of "AI hallucination" across 14

[1] © 20xx IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

TABLE III
Key points of "Hallucination" definitions within each application. The characteristics of definitions are presented in **Bold**, although they may be similar across different applications.

Application	Number of Papers	LLM Generated Key Points of Definitions
Chatbot	34	The definitions collectively highlight the central theme of AI-generated content **deviating from factual correctness**, at times even leading to entirely **fictional or erroneous information**. In essence, AI hallucination underscores the ongoing challenge of maintaining accuracy and reliability in AI-generated content within the context of chatbot applications.
Dialogue Setting	8	The definitions collectively underscore the challenge of ensuring accuracy and reliability in dialogue systems, given the potential pitfalls associated with generating content that is **unsupported, nonsensical, or factually incorrect**. These issues are particularly pertinent when deploying large pre-trained language models in dialogue applications, as they struggle with maintaining fidelity to the source material while generating coherent and accurate responses.
Generative AI	50	The definitions collectively emphasize the complexity of ensuring factual accuracy and reliability in AI-generated content within generative AI applications, highlighting the potential pitfalls of **deviating from adherence to factual correctness**.
Academia	88	A common thread among these definitions is the generation of text or content by AI models that **lacks fidelity to factual accuracy, reality, or the intended context**.
Health	82	The key idea common to all the definitions is that "AI hallucination" occurs when AI systems generate information that **deviates from factual accuracy, context, or established knowledge**. In essence, AI hallucination manifests as the production of text that, though **potentially plausible, deviates from established facts or knowledge** in health applications.
Legal and Ethical Setting	16	The definitions collectively emphasize the multifaceted challenges posed by AI hallucination in the legal and ethical context. They highlight **issues of accuracy, confidence, relevance, context, and potential misinformation**, underscoring the critical importance of addressing these challenges to ensure the responsible and ethical use of AI systems.
Science	10	Across the definitions, the central theme is that AI hallucination involves the generation of text or information that **deviates from factual accuracy, coherence, or faithfulness to the input or source content**, with potential consequences for scientific accuracy and integrity.
Technology	8	The definitions reflect the multifaceted nature of AI hallucination in technology applications, encompassing **accuracy, unpredictability, credibility, and the balance between reasonableness and correctness**.
Text Translation	4	The definitions collectively emphasize the central theme of "AI hallucination" in text translation, which revolves around challenges related to maintaining **fidelity, coherence, and relevance** in the generated translations to ensure accurate and meaningful output.
Question and Answering	7	"AI hallucination" in question and answer applications raises concerns related to the **accuracy, truthfulness, and potential spread of misinformation** in AI-generated answers, emphasizing the need for improving the reliability of these systems.
Text Summarization	19	The definitions highlight the multifaceted challenges posed by "AI hallucination" in text summarization, encompassing issues related to **fidelity, coherence, factual accuracy, and the preservation of the original meaning** in generated summaries.
Others *	7	These diverse applications collectively emphasize the challenge of maintaining **accuracy, coherence, and trustworthiness** in AI-generated content, highlighting the need for tailored approaches to address domain-specific concerns.

* Including: Investment portfolio, Journalism, Reinforcement Learning, Retail, Sport, and Survey Setting.

TABLE IV
Some popular press articles on AI hallucination

	What the Press Article Discussed…	The Real Meaning the Press Article Conveys about "AI Hallucination"	Source
1	CNBC provided some examples where ChatGPT generated outputs that sounded correct but weren't actually true, such as a legal brief written by ChatGPT to a Manhattan federal judge	When an AI model "hallucinates," it generates fabricated information in response to a user's prompt, but presents it as if it's factual and correct	CNBC [33]
2	The New York Times asked ChatGPT, Google's Brad, and Microsoft's Bing: When did The New York Times first report on "artificial intelligence"?	Chatbots provide inaccurate answers to questions; although false, the responses appear plausible as they blur and conflate people, events, and ideas	The New York Times [34]
3	The New York Times traced the evolution of the term "hallucination" throughout the newspaper's history	-	The New York Times [35]
4	CNN addressed the major issue of "AI hallucination" and narrated on the responses of OpenAI's and Google's CEOs to the question: Can hallucination be prevented?	AI-powered tools like ChatGPT impress with their ability to provide human-like responses, but a growing concern is their tendency to just make things up	CNN [36]
5	Forbes narrated the history of artificial neural networks, which started around eight decades ago, when researchers sought to replicate the functioning of the brain	"AI hallucination" refers to unrealistic ideas about achieving "artificial general intelligence" (AGI), while understanding of how our brains work is limited	Forbes [37]

more systematic, consistent and semantically nuanced terms that can replace "hallucinations" for the reasons noted here. As one step toward such a call, we presented a short summary from one of the broadest manual literature reviews on this topic to date. Our findings illustrate the current lack of consistency and consensus on this issue, but also bring to light some recent options that are good alternatives. More work is needed to develop a systematic taxonomy that can be widely adopted as we discuss these issues in the context of AI applications.

Appendix
Review of the "AI Hallucination" definitions

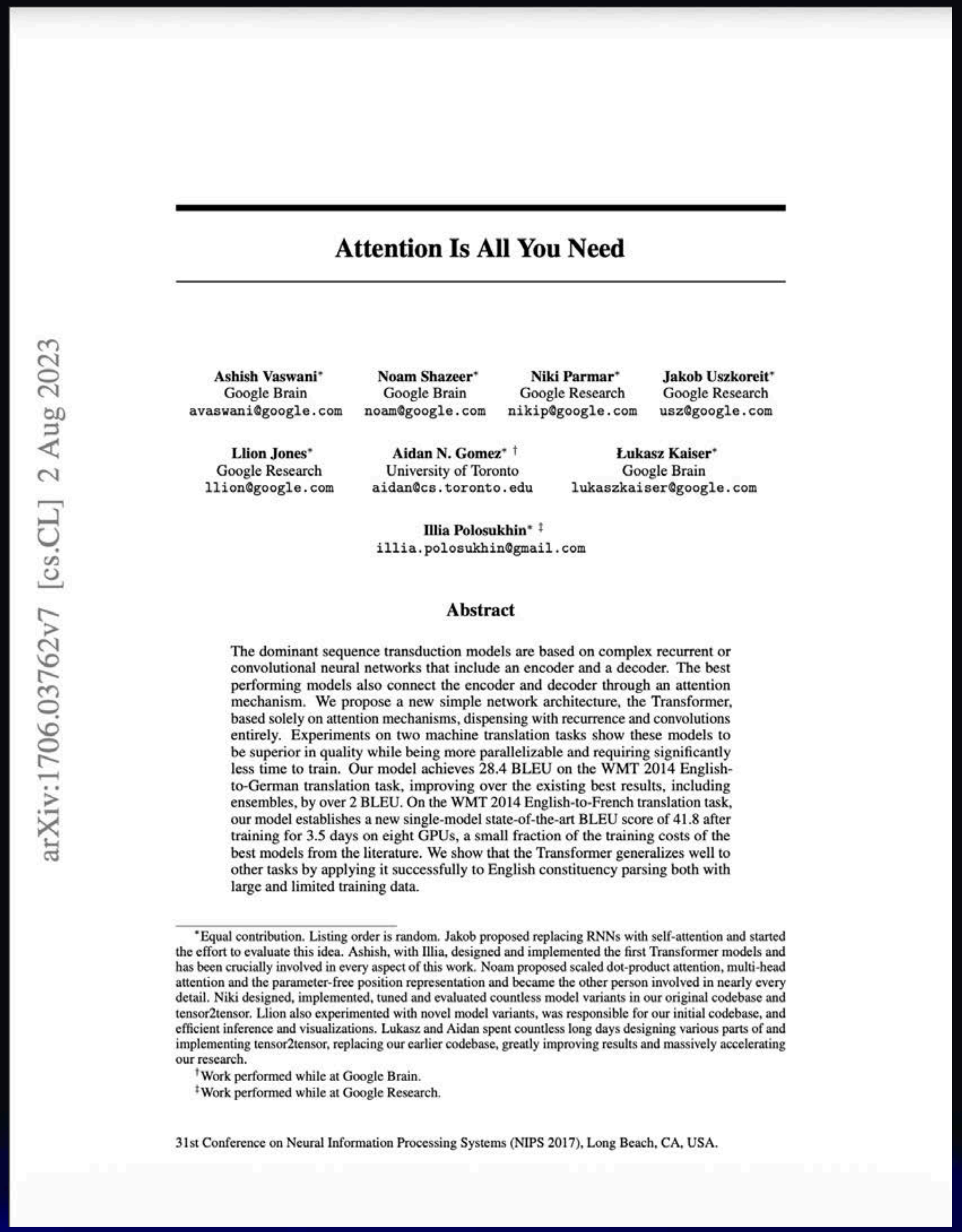
Hallucination, cont’d

1. Architectural reality:
   Hallucination emerges from the probabilistic nature of transformer models.
   “Attention Is All You Need“ by Vaswani et al. — read that!

2. Evaluation asymmetry:
   Training objectives and benchmarks reward fluency and surface correctness more than epistemic reliability.
   This can stabilize hallucination behaviour.

Quellen:

- zu 1: https://arxiv.org/abs/1706.03762
- zu 2: https://arxiv.org/abs/2512.19920



arXiv:1706.03762v7 [cs.CL] 2 Aug 2023

Attention Is All You Need

Ashish Vaswani[*]
Google Brain
avaswani@google.com

Noam Shazeer[*]
Google Brain
noam@google.com

Niki Parmar[*]
Google Research
nikip@google.com

Jakob Uszkoreit[*]
Google Research
usz@google.com

Llion Jones[*]
Google Research
llion@google.com

Aidan N. Gomez[*] [†]
University of Toronto
aidan@cs.toronto.edu

Łukasz Kaiser[*]
Google Brain
lukaszkaiser@google.com

Illia Polosukhin[*] [‡]
illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to be superior in quality while being more parallelizable and requiring significantly less time to train. Our model achieves 28.4 BLEU on the WMT 2014 English-to-German translation task, improving over the existing best results, including ensembles, by over 2 BLEU. On the WMT 2014 English-to-French translation task, our model establishes a new single-model state-of-the-art BLEU score of 41.8 after training for 3.5 days on eight GPUs, a small fraction of the training costs of the best models from the literature. We show that the Transformer generalizes well to other tasks by applying it successfully to English constituency parsing both with large and limited training data.

[*] Equal contribution. Listing order is random. Jakob proposed replacing RNNs with self-attention and started the effort to evaluate this idea. Ashish, with Illia, designed and implemented the first Transformer models and has been crucially involved in every aspect of this work. Noam proposed scaled dot-product attention, multi-head attention and the parameter-free position representation and became the other person involved in nearly every detail. Niki designed, implemented, tuned and evaluated countless model variants in our original codebase and tensor2tensor. Llion also experimented with novel model variants, was responsible for our initial codebase, and efficient inference and visualizations. Lukasz and Aidan spent countless long days designing various parts of and implementing tensor2tensor, replacing our earlier codebase, greatly improving results and massively accelerating our research.

[†] Work performed while at Google Brain.

[‡] Work performed while at Google Research.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

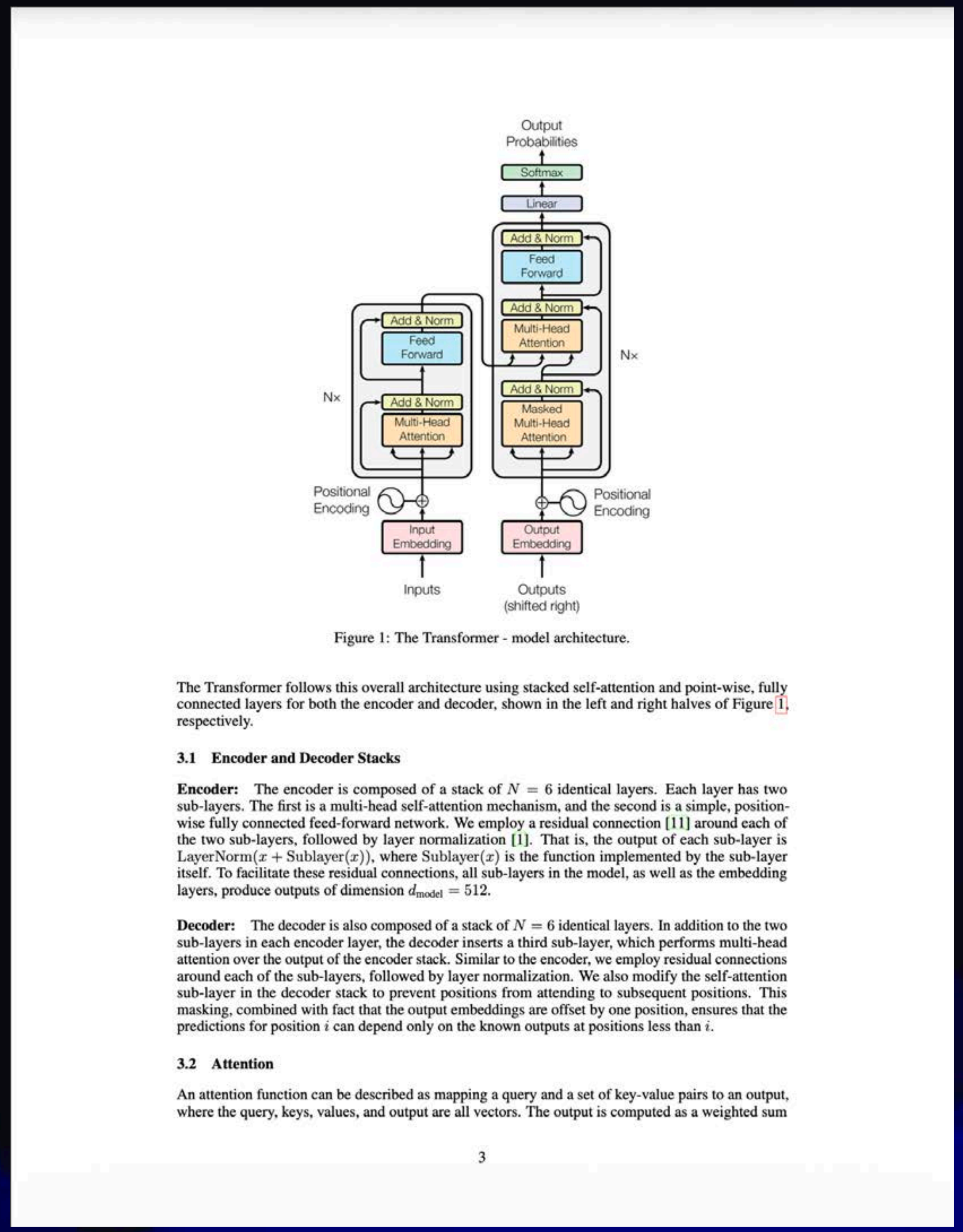


Figure 1: The Transformer - model architecture.

The Transformer follows this overall architecture using stacked self-attention and point-wise, fully connected layers for both the encoder and decoder, shown in the left and right halves of Figure 1, respectively.

3.1 Encoder and Decoder Stacks

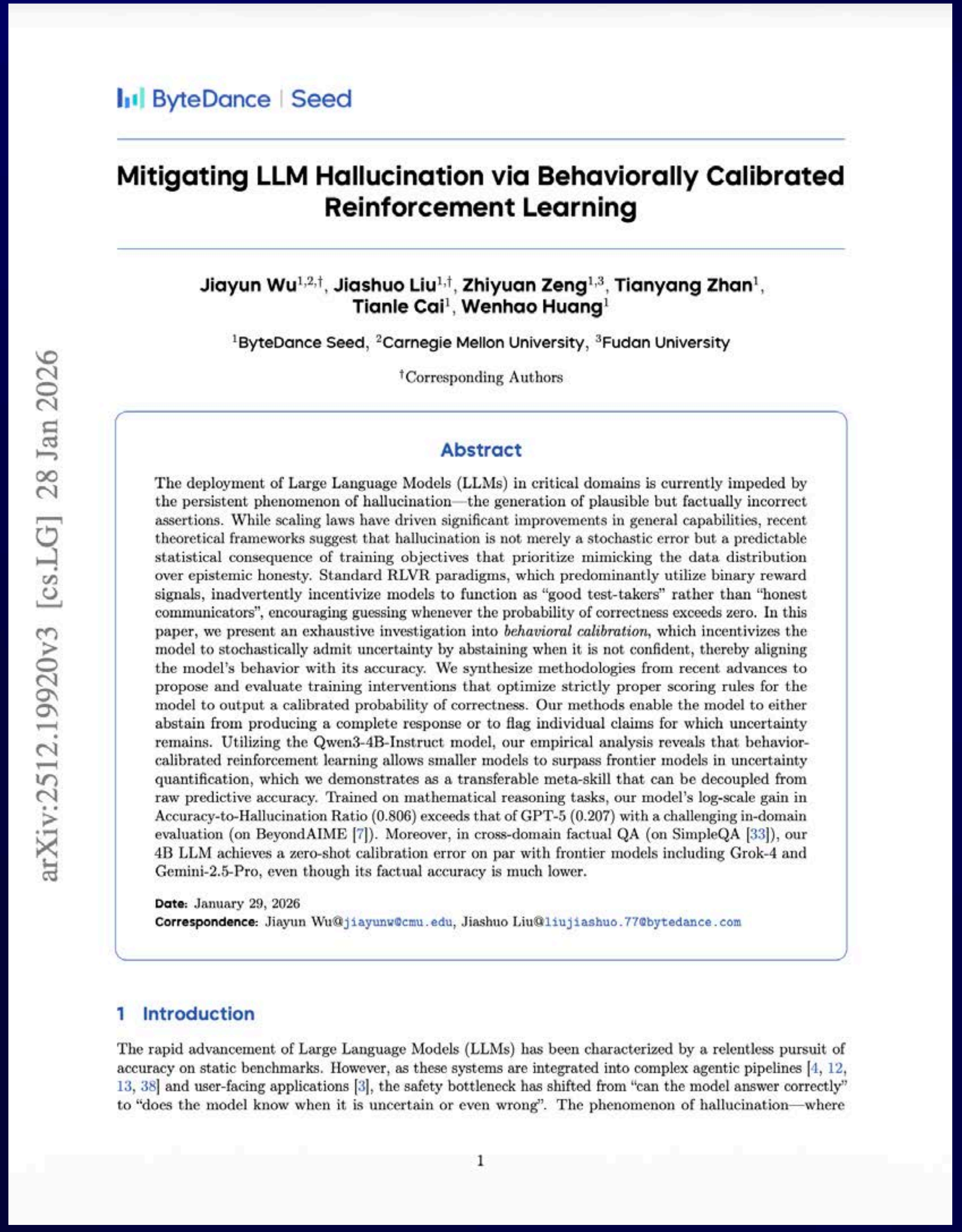
Encoder: The encoder is composed of a stack of $N = 6$ identical layers. Each layer has two sub-layers. The first is a multi-head self-attention mechanism, and the second is a simple, position-wise fully connected feed-forward network. We employ a residual connection [11] around each of the two sub-layers, followed by layer normalization [1]. That is, the output of each sub-layer is $\mathrm{LayerNorm}(x + \mathrm{Sublayer}(x))$, where $\mathrm{Sublayer}(x)$ is the function implemented by the sub-layer itself. To facilitate these residual connections, all sub-layers in the model, as well as the embedding layers, produce outputs of dimension $d_{\text{model}} = 512$.

Decoder: The decoder is also composed of a stack of $N = 6$ identical layers. In addition to the two sub-layers in each encoder layer, the decoder inserts a third sub-layer, which performs multi-head attention over the output of the encoder stack. Similar to the encoder, we employ residual connections around each of the sub-layers, followed by layer normalization. We also modify the self-attention sub-layer in the decoder stack to prevent positions from attending to subsequent positions. This masking, combined with fact that the output embeddings are offset by one position, ensures that the predictions for position i can depend only on the known outputs at positions less than i.

3.2 Attention

An attention function can be described as mapping a query and a set of key-value pairs to an output, where the query, keys, values, and output are all vectors. The output is computed as a weighted sum

3



ByteDance | Seed

arXiv:2512.19920v3 [cs.LG] 28 Jan 2026

Mitigating LLM Hallucination via Behaviorally Calibrated Reinforcement Learning

Jiayun Wu[1,2,†], Jiashuo Liu[1,†], Zhiyuan Zeng[1,3], Tianyang Zhan[1], Tianle Cai[1], Wenhao Huang[1]

[1]ByteDance Seed, [2]Carnegie Mellon University, [3]Fudan University

[†]Corresponding Authors

Abstract

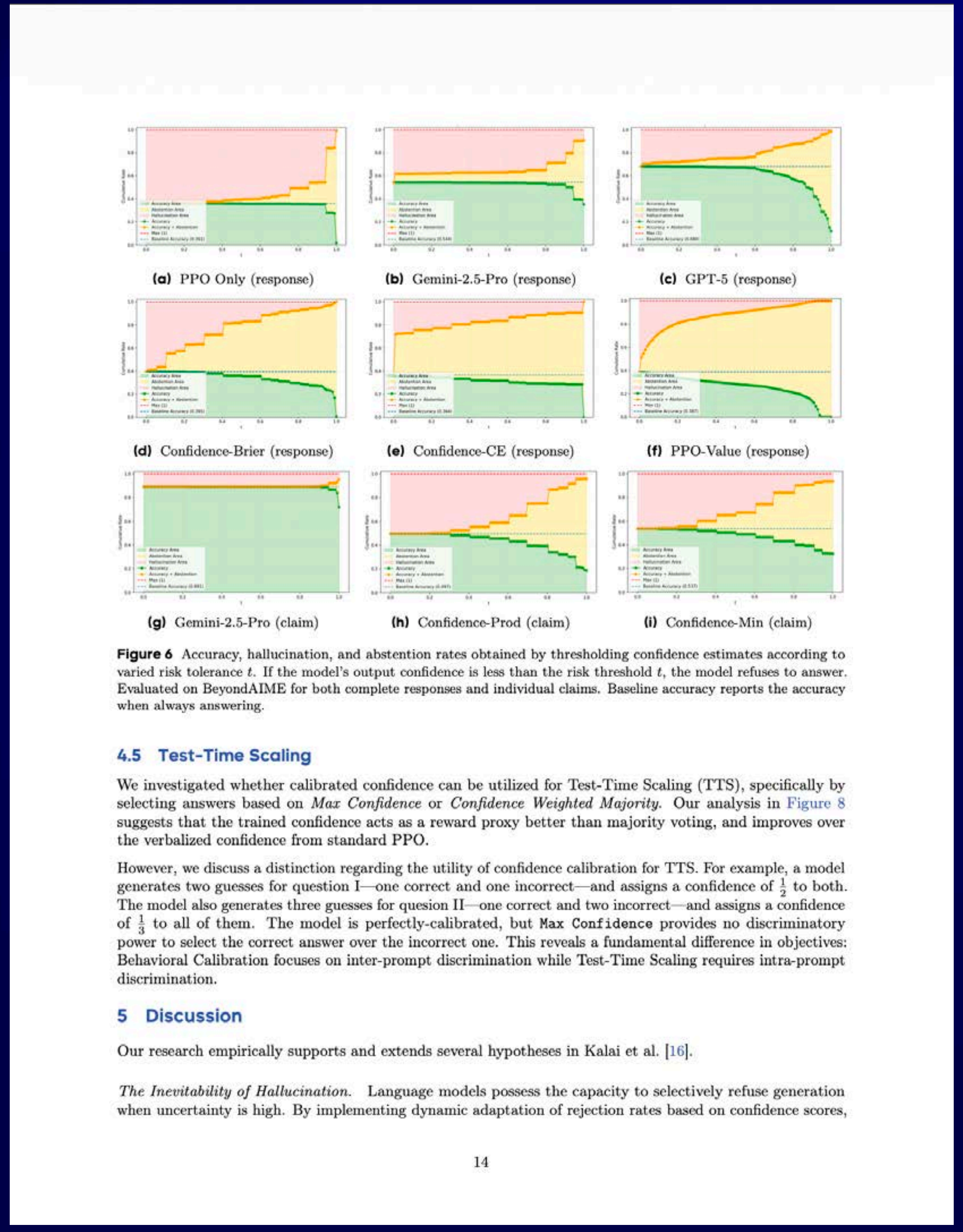
The deployment of Large Language Models (LLMs) in critical domains is currently impeded by the persistent phenomenon of hallucination—the generation of plausible but factually incorrect assertions. While scaling laws have driven significant improvements in general capabilities, recent theoretical frameworks suggest that hallucination is not merely a stochastic error but a predictable statistical consequence of training objectives that prioritize mimicking the data distribution over epistemic honesty. Standard RLVR paradigms, which predominantly utilize binary reward signals, inadvertently incentivize models to function as “good test-takers” rather than “honest communicators”, encouraging guessing whenever the probability of correctness exceeds zero. In this paper, we present an exhaustive investigation into *behavioral calibration*, which incentivizes the model to stochastically admit uncertainty by abstaining when it is not confident, thereby aligning the model’s behavior with its accuracy. We synthesize methodologies from recent advances to propose and evaluate training interventions that optimize strictly proper scoring rules for the model to output a calibrated probability of correctness. Our methods enable the model to either abstain from producing a complete response or to flag individual claims for which uncertainty remains. Utilizing the Qwen3-4B-Instruct model, our empirical analysis reveals that behavior-calibrated reinforcement learning allows smaller models to surpass frontier models in uncertainty quantification, which we demonstrates as a transferable meta-skill that can be decoupled from raw predictive accuracy. Trained on mathematical reasoning tasks, our model’s log-scale gain in Accuracy-to-Hallucination Ratio (0.806) exceeds that of GPT-5 (0.207) with a challenging in-domain evaluation (on BeyondAIME [7]). Moreover, in cross-domain factual QA (on SimpleQA [33]), our 4B LLM achieves a zero-shot calibration error on par with frontier models including Grok-4 and Gemini-2.5-Pro, even though its factual accuracy is much lower.

Date: January 29, 2026
Correspondence: Jiayun Wu@jiayunw@cmu.edu, Jiashuo Liu@liujiashuo.77@bytedance.com

1 Introduction

The rapid advancement of Large Language Models (LLMs) has been characterized by a relentless pursuit of accuracy on static benchmarks. However, as these systems are integrated into complex agentic pipelines [4, 12, 13, 38] and user-facing applications [3], the safety bottleneck has shifted from “can the model answer correctly” to “does the model know when it is uncertain or even wrong”. The phenomenon of hallucination—where

1



(a) PPO Only (response) (b) Gemini-2.5-Pro (response) (c) GPT-5 (response)

(d) Confidence-Brier (response) (e) Confidence-CE (response) (f) PPO-Value (response)

(g) Gemini-2.5-Pro (claim) (h) Confidence-Prod (claim) (i) Confidence-Min (claim)

Figure 6 Accuracy, hallucination, and abstention rates obtained by thresholding confidence estimates according to varied risk tolerance t. If the model’s output confidence is less than the risk threshold t, the model refuses to answer. Evaluated on BeyondAIME for both complete responses and individual claims. Baseline accuracy reports the accuracy when always answering.

4.5 Test-Time Scaling

We investigated whether calibrated confidence can be utilized for Test-Time Scaling (TTS), specifically by selecting answers based on *Max Confidence* or *Confidence Weighted Majority*. Our analysis in Figure 8 suggests that the trained confidence acts as a reward proxy better than majority voting, and improves over the verbalized confidence from standard PPO.

However, we discuss a distinction regarding the utility of confidence calibration for TTS. For example, a model generates two guesses for question I—one correct and one incorrect—and assigns a confidence of $\frac{1}{2}$ to both. The model also generates three guesses for quesion II—one correct and two incorrect—and assigns a confidence of $\frac{1}{3}$ to all of them. The model is perfectly-calibrated, but `Max Confidence` provides no discriminatory power to select the correct answer over the incorrect one. This reveals a fundamental difference in objectives: Behavioral Calibration focuses on inter-prompt discrimination while Test-Time Scaling requires intra-prompt discrimination.

5 Discussion

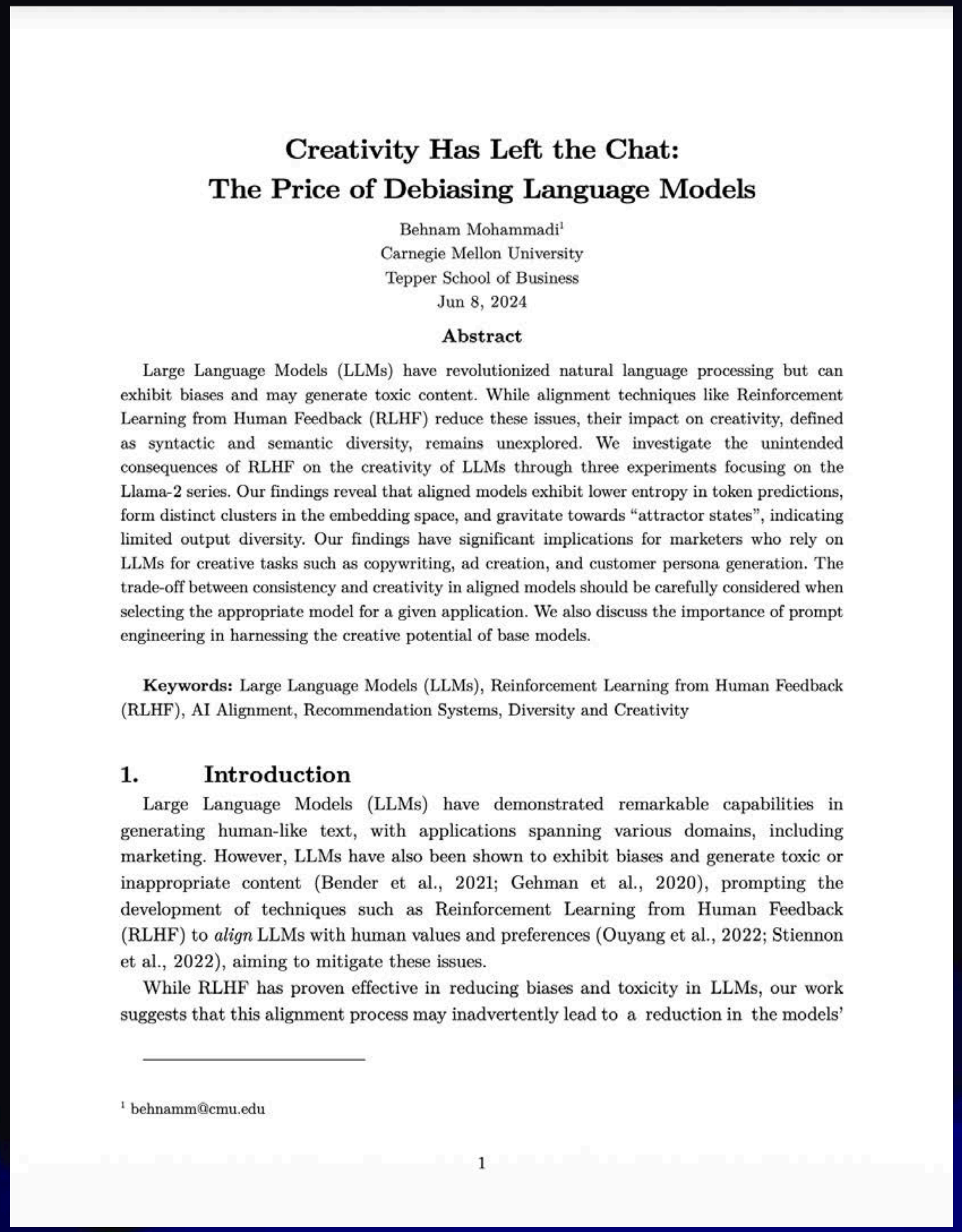
Our research empirically supports and extends several hypotheses in Kalai et al. [16].

The Inevitability of Hallucination. Language models possess the capacity to selectively refuse generation when uncertainty is high. By implementing dynamic adaptation of rejection rates based on confidence scores,

14

Trust Is a Trade-off

- A model that can never be wrong must never speculate
  — it would be a database or an expert system,
  not a language model.
- A model that generates creatively must be allowed to be wrong occasionally
  — this is a design principle, not a failure.
- Optimization under conflicting objectives always implies a trade-off
  — gain in one dimension entails loss in another.
- You cannot demand “fundamentally reliable AI” without understanding this trade-off
  — you would be demanding a contradiction.
- There is no purely technical solution
  — only conscious, transparent choices about trade-offs.



Creativity Has Left the Chat: The Price of Debiasing Language Models

Behnam Mohammadi[1]
Carnegie Mellon University
Tepper School of Business
Jun 8, 2024

Abstract

Large Language Models (LLMs) have revolutionized natural language processing but can exhibit biases and may generate toxic content. While alignment techniques like Reinforcement Learning from Human Feedback (RLHF) reduce these issues, their impact on creativity, defined as syntactic and semantic diversity, remains unexplored. We investigate the unintended consequences of RLHF on the creativity of LLMs through three experiments focusing on the Llama-2 series. Our findings reveal that aligned models exhibit lower entropy in token predictions, form distinct clusters in the embedding space, and gravitate towards “attractor states”, indicating limited output diversity. Our findings have significant implications for marketers who rely on LLMs for creative tasks such as copywriting, ad creation, and customer persona generation. The trade-off between consistency and creativity in aligned models should be carefully considered when selecting the appropriate model for a given application. We also discuss the importance of prompt engineering in harnessing the creative potential of base models.

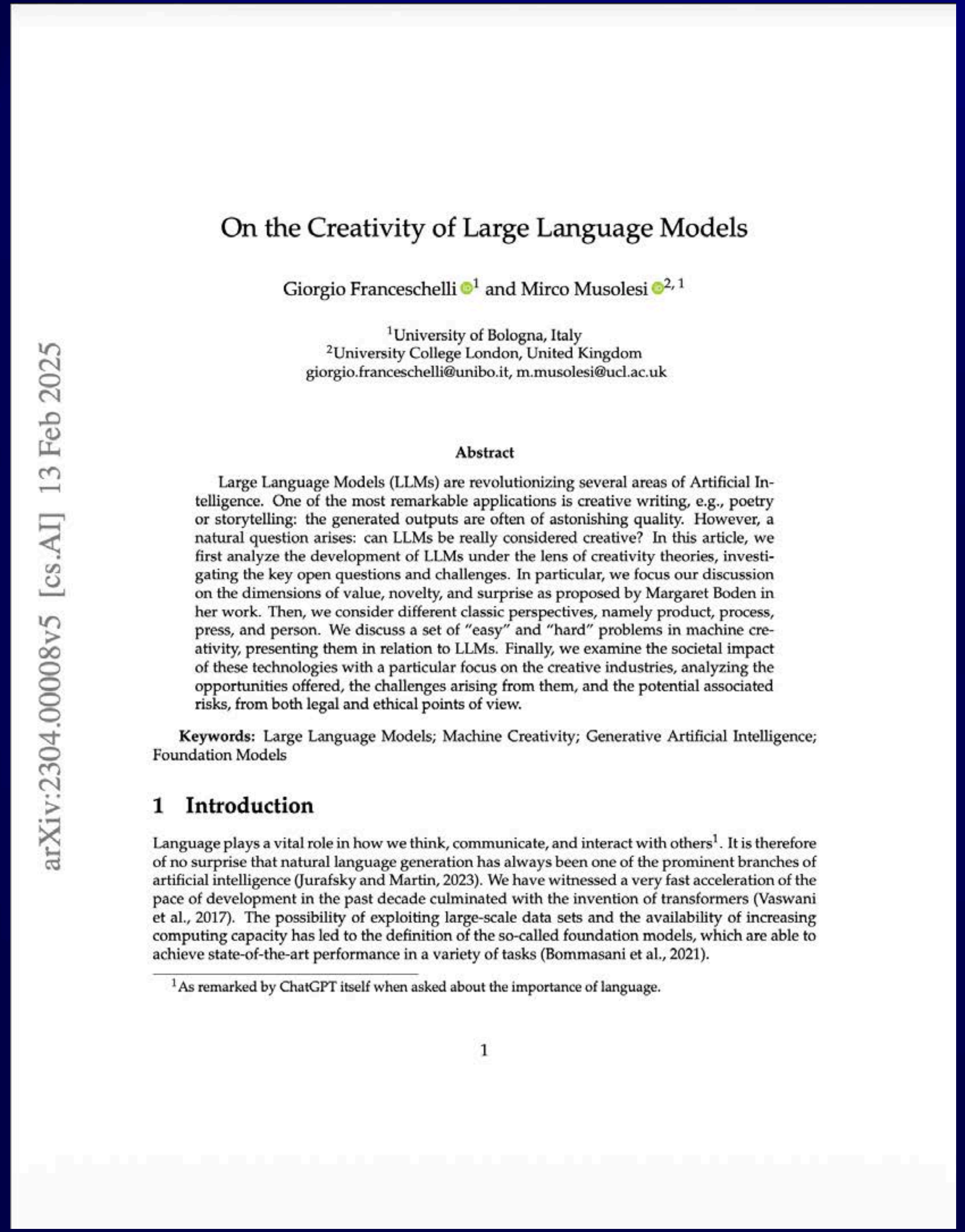
Keywords: Large Language Models (LLMs), Reinforcement Learning from Human Feedback (RLHF), AI Alignment, Recommendation Systems, Diversity and Creativity

1. Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities in generating human-like text, with applications spanning various domains, including marketing. However, LLMs have also been shown to exhibit biases and generate toxic or inappropriate content (Bender et al., 2021; Gehman et al., 2020), prompting the development of techniques such as Reinforcement Learning from Human Feedback (RLHF) to *align* LLMs with human values and preferences (Ouyang et al., 2022; Stiennon et al., 2022), aiming to mitigate these issues.

While RLHF has proven effective in reducing biases and toxicity in LLMs, our work suggests that this alignment process may inadvertently lead to a reduction in the models’

[1] behnamm@cmu.edu



On the Creativity of Large Language Models

Giorgio Franceschelli [1] and Mirco Musolesi [2, 1]

[1]University of Bologna, Italy
[2]University College London, United Kingdom
giorgio.franceschelli@unibo.it, m.musolesi@ucl.ac.uk

Abstract

Large Language Models (LLMs) are revolutionizing several areas of Artificial Intelligence. One of the most remarkable applications is creative writing, e.g., poetry or storytelling: the generated outputs are often of astonishing quality. However, a natural question arises: can LLMs be really considered creative? In this article, we first analyze the development of LLMs under the lens of creativity theories, investigating the key open questions and challenges. In particular, we focus our discussion on the dimensions of value, novelty, and surprise as proposed by Margaret Boden in her work. Then, we consider different classic perspectives, namely product, process, press, and person. We discuss a set of ”easy” and “hard” problems in machine creativity, presenting them in relation to LLMs. Finally, we examine the societal impact of these technologies with a particular focus on the creative industries, analyzing the opportunities offered, the challenges arising from them, and the potential associated risks, from both legal and ethical points of view.

Keywords: Large Language Models; Machine Creativity; Generative Artificial Intelligence; Foundation Models

1 Introduction

Language plays a vital role in how we think, communicate, and interact with others[1]. It is therefore of no surprise that natural language generation has always been one of the prominent branches of artificial intelligence (Jurafsky and Martin, 2023). We have witnessed a very fast acceleration of the pace of development in the past decade culminated with the invention of transformers (Vaswani et al., 2017). The possibility of exploiting large-scale data sets and the availability of increasing computing capacity has led to the definition of the so-called foundation models, which are able to achieve state-of-the-art performance in a variety of tasks (Bommasani et al., 2021).

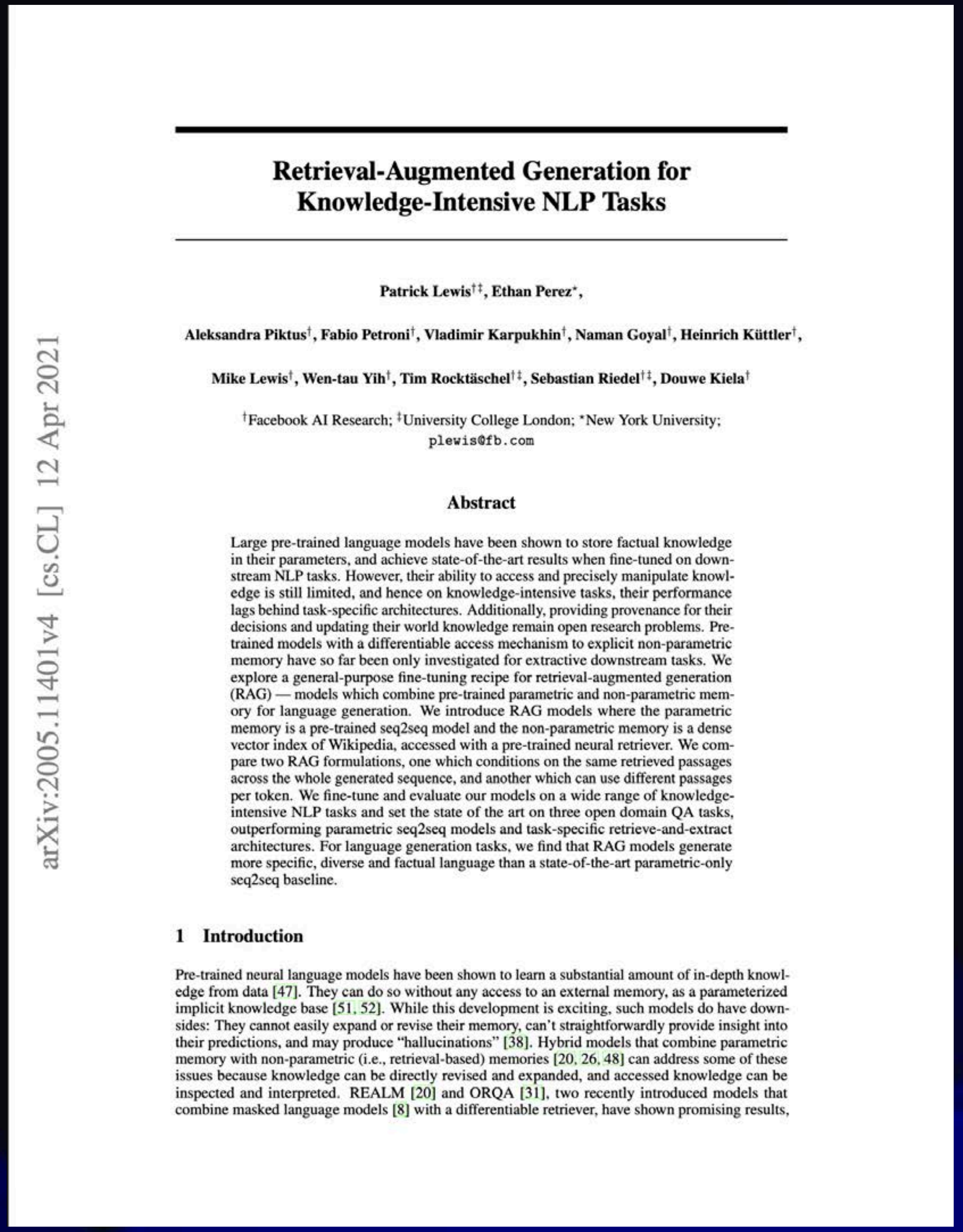
[1] As remarked by ChatGPT itself when asked about the importance of language.

Approaches to Trust

Spoiler: Trust is not a property of the model —
it is a property of the architecture

RAG:
Retrieval-Augmented Generation

- Even before ChatGPT, Meta addresses the problem of ***knowledge-intensive tasks***
  - Dense factual knowledge is only weakly encoded in model weights.
  - Pre-training is expensive → model knowledge is always outdated
  - Idea: inject fresh, exact knowledge at inference time
  - External embedding vector database → knowledge store
  - Major progress — but new failure modes emerge
    contradictions, chunking errors, lost-in-the-middle, multi-hop limits, latency, etc.
  - Reduces hallucinations — a meaningful step towards trust



arXiv:2005.11401v4 [cs.CL] 12 Apr 2021

Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks

Patrick Lewis[†‡], Ethan Perez[*],

Aleksandra Piktus[†], Fabio Petroni[†], Vladimir Karpukhin[†], Naman Goyal[†], Heinrich Küttler[†],

Mike Lewis[†], Wen-tau Yih[†], Tim Rocktäschel[†‡], Sebastian Riedel[†‡], Douwe Kiela[†]

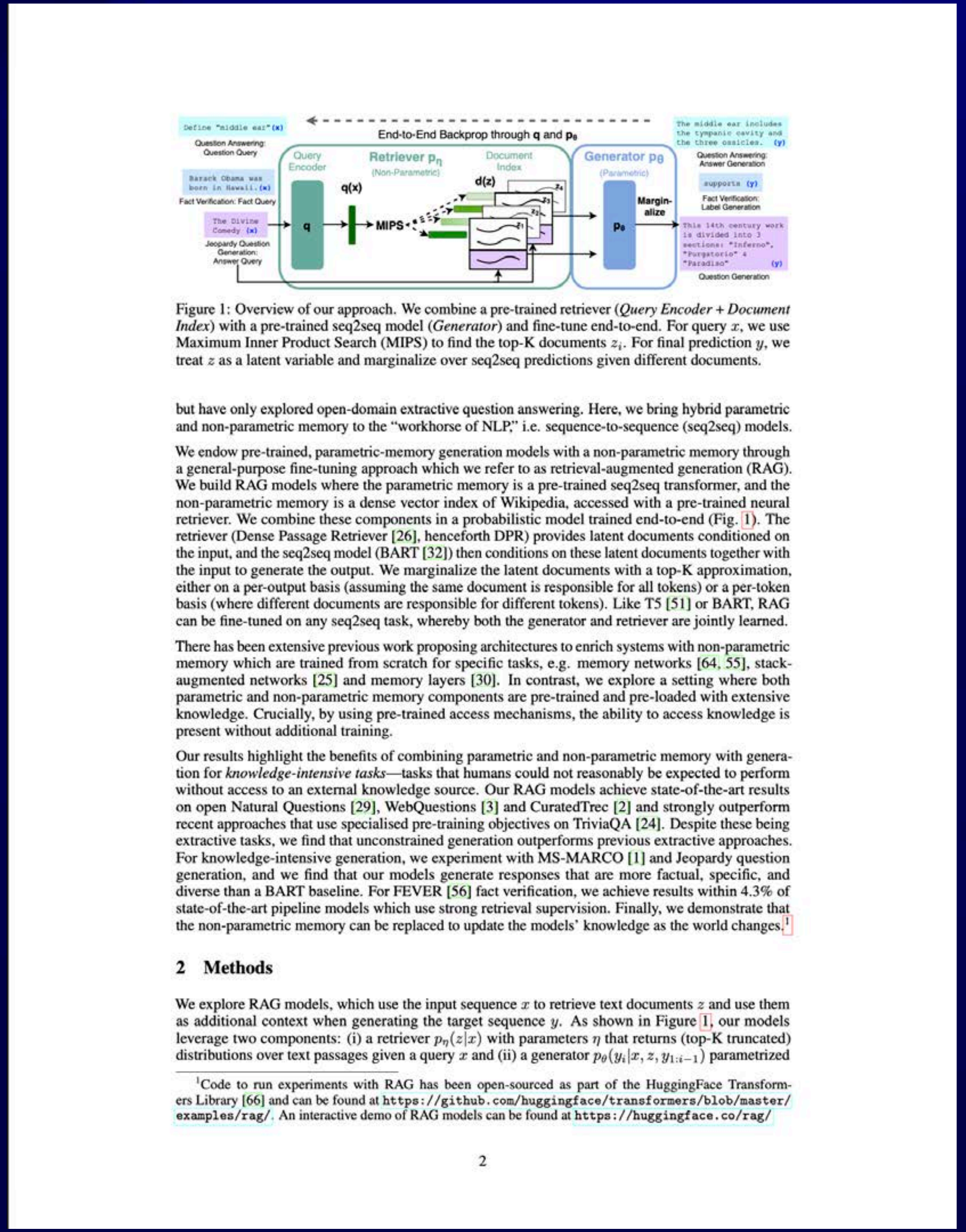
[†]Facebook AI Research; [‡]University College London; [*]New York University;
plewis@fb.com

Abstract

Large pre-trained language models have been shown to store factual knowledge in their parameters, and achieve state-of-the-art results when fine-tuned on downstream NLP tasks. However, their ability to access and precisely manipulate knowledge is still limited, and hence on knowledge-intensive tasks, their performance lags behind task-specific architectures. Additionally, providing provenance for their decisions and updating their world knowledge remain open research problems. Pre-trained models with a differentiable access mechanism to explicit non-parametric memory have so far been only investigated for extractive downstream tasks. We explore a general-purpose fine-tuning recipe for retrieval-augmented generation (RAG) — models which combine pre-trained parametric and non-parametric memory for language generation. We introduce RAG models where the parametric memory is a pre-trained seq2seq model and the non-parametric memory is a dense vector index of Wikipedia, accessed with a pre-trained neural retriever. We compare two RAG formulations, one which conditions on the same retrieved passages across the whole generated sequence, and another which can use different passages per token. We fine-tune and evaluate our models on a wide range of knowledge-intensive NLP tasks and set the state of the art on three open domain QA tasks, outperforming parametric seq2seq models and task-specific retrieve-and-extract architectures. For language generation tasks, we find that RAG models generate more specific, diverse and factual language than a state-of-the-art parametric-only seq2seq baseline.

1 Introduction

Pre-trained neural language models have been shown to learn a substantial amount of in-depth knowledge from data [47]. They can do so without any access to an external memory, as a parameterized implicit knowledge base [51, 52]. While this development is exciting, such models do have downsides: They cannot easily expand or revise their memory, can't straightforwardly provide insight into their predictions, and may produce "hallucinations" [38]. Hybrid models that combine parametric memory with non-parametric (i.e., retrieval-based) memories [20, 26, 48] can address some of these issues because knowledge can be directly revised and expanded, and accessed knowledge can be inspected and interpreted. REALM [20] and ORQA [31], two recently introduced models that combine masked language models [8] with a differentiable retriever, have shown promising results,



Define "middle ear" (x)
Question Answering: Question Query
Barack Obama was born in Hawaii. (x)
Fact Verification: Fact Query
The Divine Comedy (x)
Jeopardy Question Generation: Answer Query
End-to-End Backprop through q and pθ
Query Encoder
Retriever pη
(Non-Parametric)
Document Index
q
q(x)
MIPS
d(z)
z1
z2
z3
z4
Generator pθ
(Parametric)
pθ
Margin-alize
The middle ear includes the tympanic cavity and the three ossicles. (y)
Question Answering: Answer Generation
supports (y)
Fact Verification: Label Generation
This 14th century work is divided into 3 sections: "Inferno", "Purgatorio" & "Paradiso" (y)
Question Generation

Figure 1: Overview of our approach. We combine a pre-trained retriever (*Query Encoder* + *Document Index*) with a pre-trained seq2seq model (*Generator*) and fine-tune end-to-end. For query x, we use Maximum Inner Product Search (MIPS) to find the top-K documents z_i. For final prediction y, we treat z as a latent variable and marginalize over seq2seq predictions given different documents.

but have only explored open-domain extractive question answering. Here, we bring hybrid parametric and non-parametric memory to the "workhorse of NLP," i.e. sequence-to-sequence (seq2seq) models.

We endow pre-trained, parametric-memory generation models with a non-parametric memory through a general-purpose fine-tuning approach which we refer to as retrieval-augmented generation (RAG). We build RAG models where the parametric memory is a pre-trained seq2seq transformer, and the non-parametric memory is a dense vector index of Wikipedia, accessed with a pre-trained neural retriever. We combine these components in a probabilistic model trained end-to-end (Fig. 1). The retriever (Dense Passage Retriever [26], henceforth DPR) provides latent documents conditioned on the input, and the seq2seq model (BART [32]) then conditions on these latent documents together with the input to generate the output. We marginalize the latent documents with a top-K approximation, either on a per-output basis (assuming the same document is responsible for all tokens) or a per-token basis (where different documents are responsible for different tokens). Like T5 [51] or BART, RAG can be fine-tuned on any seq2seq task, whereby both the generator and retriever are jointly learned.

There has been extensive previous work proposing architectures to enrich systems with non-parametric memory which are trained from scratch for specific tasks, e.g. memory networks [64, 55], stack-augmented networks [25] and memory layers [30]. In contrast, we explore a setting where both parametric and non-parametric memory components are pre-trained and pre-loaded with extensive knowledge. Crucially, by using pre-trained access mechanisms, the ability to access knowledge is present without additional training.

Our results highlight the benefits of combining parametric and non-parametric memory with generation for *knowledge-intensive tasks*—tasks that humans could not reasonably be expected to perform without access to an external knowledge source. Our RAG models achieve state-of-the-art results on open Natural Questions [29], WebQuestions [3] and CuratedTrec [2] and strongly outperform recent approaches that use specialised pre-training objectives on TriviaQA [24]. Despite these being extractive tasks, we find that unconstrained generation outperforms previous extractive approaches. For knowledge-intensive generation, we experiment with MS-MARCO [1] and Jeopardy question generation, and we find that our models generate responses that are more factual, specific, and diverse than a BART baseline. For FEVER [56] fact verification, we achieve results within 4.3% of state-of-the-art pipeline models which use strong retrieval supervision. Finally, we demonstrate that the non-parametric memory can be replaced to update the models' knowledge as the world changes.[1]

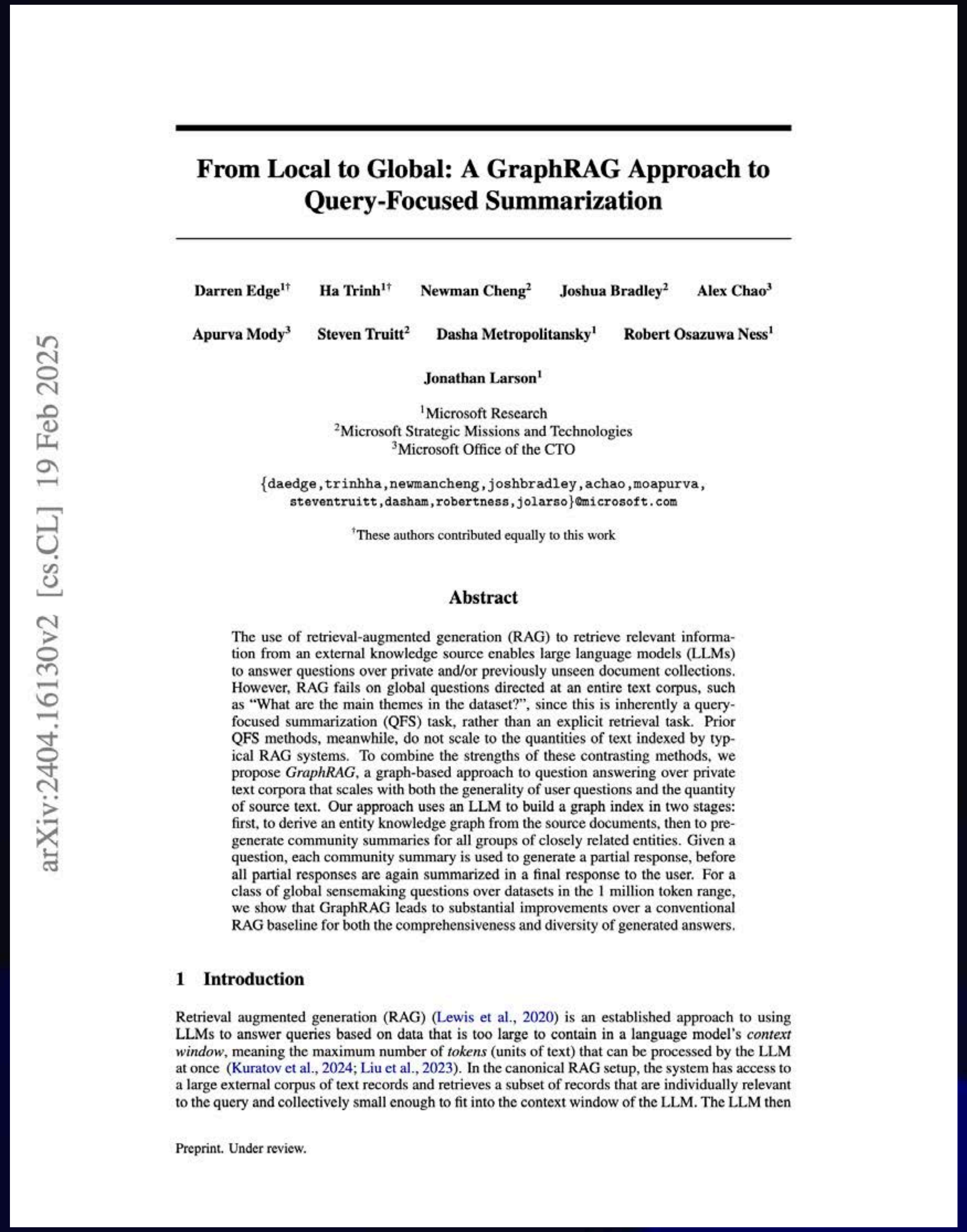
2 Methods

We explore RAG models, which use the input sequence x to retrieve text documents z and use them as additional context when generating the target sequence y. As shown in Figure 1, our models leverage two components: (i) a retriever $p_\eta(z|x)$ with parameters η that returns (top-K truncated) distributions over text passages given a query x and (ii) a generator $p_\theta(y_i|x, z, y_{1:i-1})$ parametrized

[1] Code to run experiments with RAG has been open-sourced as part of the HuggingFace Transformers Library [66] and can be found at https://github.com/huggingface/transformers/blob/master/examples/rag/. An interactive demo of RAG models can be found at https://huggingface.co/rag/

Knowledge Graph: KG-RAG and GraphRAG

- Special form of RAG, effective for structured information
  - Entities and their relations are encoded explicitly
  - Graph *G*=(*V*,*E*) of vertices and edges
  - Key advantage: consistency is no longer emergent — it is engineered
- Knowledge Graphs are older that RAG or LLM
  - 1960s to 1970s AI research: Semantic Networks
  - Since 2012: Google's Knowledge Graph marked the end of matrix-based internet search engines
- New bottleneck: Excellent results — but expensive.
  - Domain-bound, hard to scale and generalize.
  - Architectural mismatch with GPUs (non-SIMD, irregular memory access)



arXiv:2404.16130v2 [cs.CL] 19 Feb 2025

From Local to Global: A GraphRAG Approach to Query-Focused Summarization

Darren Edge[1†] Ha Trinh[1†] Newman Cheng[2] Joshua Bradley[2] Alex Chao[3]
Apurva Mody[3] Steven Truitt[2] Dasha Metropolitansky[1] Robert Osazuwa Ness[1]
Jonathan Larson[1]

[1]Microsoft Research
[2]Microsoft Strategic Missions and Technologies
[3]Microsoft Office of the CTO

{daedge,trinhha,newmancheng,joshbradley,achao,moapurva,
steventruitt,dasham,robertness,jolarso}@microsoft.com

[†]These authors contributed equally to this work

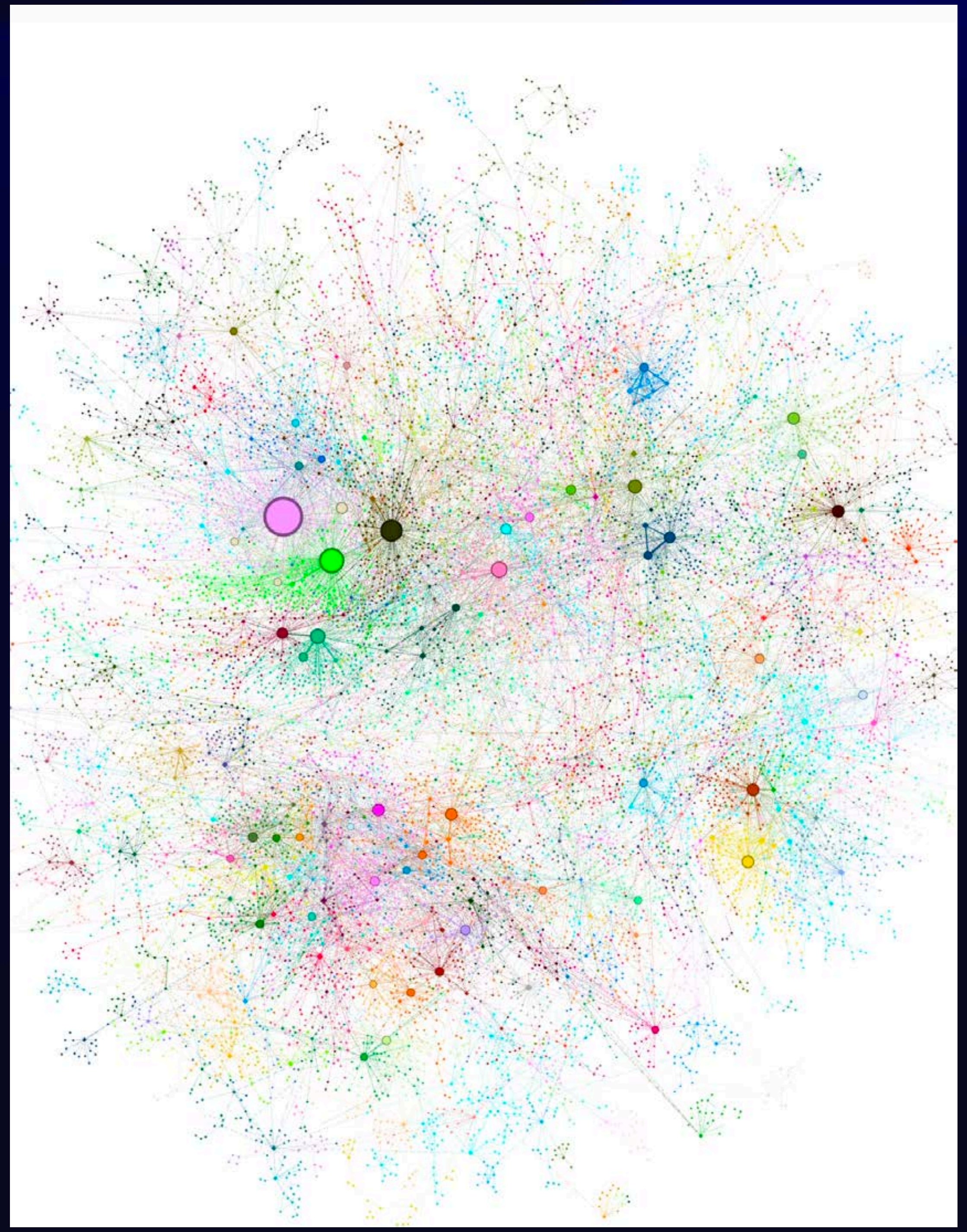
Abstract

The use of retrieval-augmented generation (RAG) to retrieve relevant information from an external knowledge source enables large language models (LLMs) to answer questions over private and/or previously unseen document collections. However, RAG fails on global questions directed at an entire text corpus, such as "What are the main themes in the dataset?", since this is inherently a query-focused summarization (QFS) task, rather than an explicit retrieval task. Prior QFS methods, meanwhile, do not scale to the quantities of text indexed by typical RAG systems. To combine the strengths of these contrasting methods, we propose *GraphRAG*, a graph-based approach to question answering over private text corpora that scales with both the generality of user questions and the quantity of source text. Our approach uses an LLM to build a graph index in two stages: first, to derive an entity knowledge graph from the source documents, then to pregenerate community summaries for all groups of closely related entities. Given a question, each community summary is used to generate a partial response, before all partial responses are again summarized in a final response to the user. For a class of global sensemaking questions over datasets in the 1 million token range, we show that GraphRAG leads to substantial improvements over a conventional RAG baseline for both the comprehensiveness and diversity of generated answers.

1 Introduction

Retrieval augmented generation (RAG) (Lewis et al., 2020) is an established approach to using LLMs to answer queries based on data that is too large to contain in a language model's *context window*, meaning the maximum number of *tokens* (units of text) that can be processed by the LLM at once (Kuratov et al., 2024; Liu et al., 2023). In the canonical RAG setup, the system has access to a large external corpus of text records and retrieves a subset of records that are individually relevant to the query and collectively small enough to fit into the context window of the LLM. The LLM then

Preprint. Under review.



The World Model of Yann LeCun

- Yann LeCun, Turing Award 2018, etc.
    - Professor at New York University, Inventor of CNNs, Head of AI at Meta for 12 years, now founding his own startup AMI Labs (Advanced Machine Intelligence)
- “… LLMs are a dead end (when it comes to superintelligence)“
- LeCuns Approach: **World Models**
    - Learned representations of physical reality
    - So very complete and continuous that they can predict consequences of actions
    - Intelligence = **grounded** prediction + planning
    - Not an extension of LLMs → A different paradigm
- Language models predict text. World models predict reality.

A Path Towards Autonomous Machine Intelligence
Version 0.9.2, 2022-06-27

Yann LeCun
Courant Institute of Mathematical Sciences, New York University yann@cs.nyu.edu
Meta - Fundamental AI Research yann@fb.com

June 27, 2022

Abstract

How could machines learn as efficiently as humans and animals? How could machines learn to reason and plan? How could machines learn representations of percepts and action plans at multiple levels of abstraction, enabling them to reason, predict, and plan at multiple time horizons? This position paper proposes an architecture and training paradigms with which to construct autonomous intelligent agents. It combines concepts such as configurable predictive world model, behavior driven through intrinsic motivation, and hierarchical joint embedding architectures trained with self-supervised learning.

Keywords: Artificial Intelligence, Machine Common Sense, Cognitive Architecture, Deep Learning, Self-Supervised Learning, Energy-Based Model, World Models, Joint Embedding Architecture, Intrinsic Motivation.

1 Prologue

This document is not a technical nor scholarly paper in the traditional sense, but a position paper expressing my vision for a path towards intelligent machines that learn more like animals and humans, that can reason and plan, and whose behavior is driven by intrinsic objectives, rather than by hard-wired programs, external supervision, or external rewards. Many ideas described in this paper (almost all of them) have been formulated by many authors in various contexts in various form. The present piece does not claim priority for any of them but presents a proposal for how to assemble them into a consistent whole. In particular, the piece pinpoints the challenges ahead. It also lists a number of avenues that are likely or unlikely to succeed.

The text is written with as little jargon as possible, and using as little mathematical prior knowledge as possible, so as to appeal to readers with a wide variety of backgrounds including neuroscience, cognitive science, and philosophy, in addition to machine learning, robotics, and other fields of engineering. I hope that this piece will help contextualize some of the research in AI whose relevance is sometimes difficult to see.

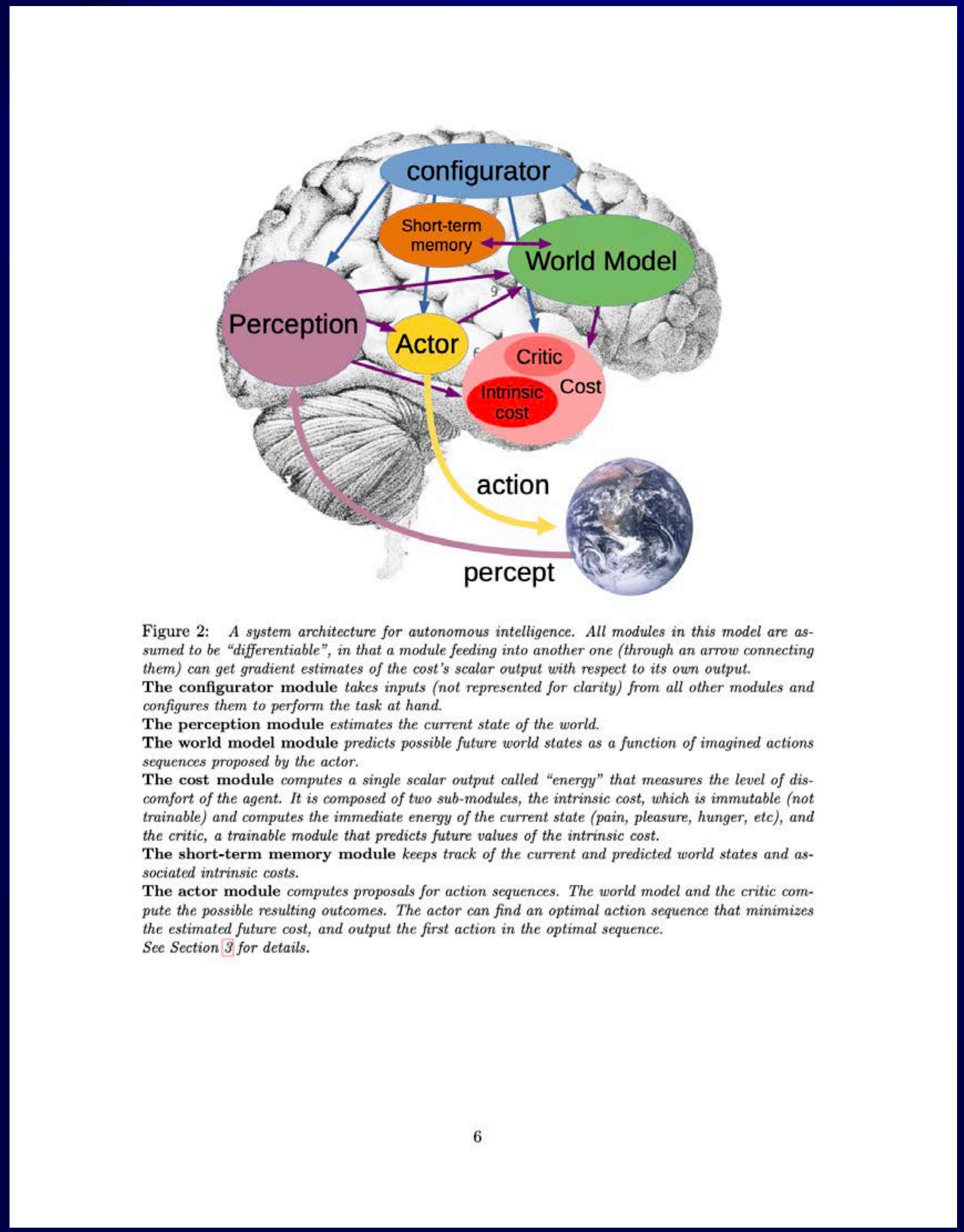


Figure 2: *A system architecture for autonomous intelligence. All modules in this model are assumed to be “differentiable”, in that a module feeding into another one (through an arrow connecting them) can get gradient estimates of the cost’s scalar output with respect to its own output.*
The configurator module *takes inputs (not represented for clarity) from all other modules and configures them to perform the task at hand.*
The perception module *estimates the current state of the world.*
The world model module *predicts possible future world states as a function of imagined actions sequences proposed by the actor.*
The cost module *computes a single scalar output called “energy” that measures the level of discomfort of the agent. It is composed of two sub-modules, the intrinsic cost, which is immutable (not trainable) and computes the immediate energy of the current state (pain, pleasure, hunger, etc), and the critic, a trainable module that predicts future values of the intrinsic cost.*
The short-term memory module *keeps track of the current and predicted world states and associated intrinsic costs.*
The actor module *computes proposals for action sequences. The world model and the critic compute the possible resulting outcomes. The actor can find an optimal action sequence that minimizes the estimated future cost, and output the first action in the optimal sequence.*
See Section 3 for details.

Trust-Engineering along the Vector of Harmfulness

- Bias → remove the asymmetries in objective functions
- Curation side-effects → restore ambiguity and balance
- Factuality → enforce epistemic constraints

“Problems cannot be solved with the same thinking that created them”, often attributed to Einstein

But this time — it’s exactly where you have to start!

Trust is not a feature.
It is a systems property.

MCP

The leading agentic protocol places the
Human-in-the-Loop
as an core element of architecture.

Die Welt ist voller Menschen, die unter den Folgen ihres ungelebten Lebens leiden.

Sie werden verbittert, kritisch oder unnachgiebig, nicht, weil die Welt zu grausam zu Ihnen ist, sondern weil sie **ihre inneren Möglichkeiten verraten** haben.

Sie sehen im Außen Feinde, die eigentlich in ihrem eigenen Inneren als **ungenutzte Talente** und unterdrückte Wunsche schlummern.

Erst wenn der Mensch beginnt, sein **eigenes Licht nicht mehr unter den Scheffel zu stellen** und die Verantwortung für sein eigenes Werden übernimmt, löst sich die Bitterkeit auf und macht dem Frieden Platz.

Carl Gustav Jung,
1875 - 1961,
founder of the school of analytical psychology

AI for autonomous Weapons?!

Step up and take responsibility!

ATARI

River Raid

Tournament