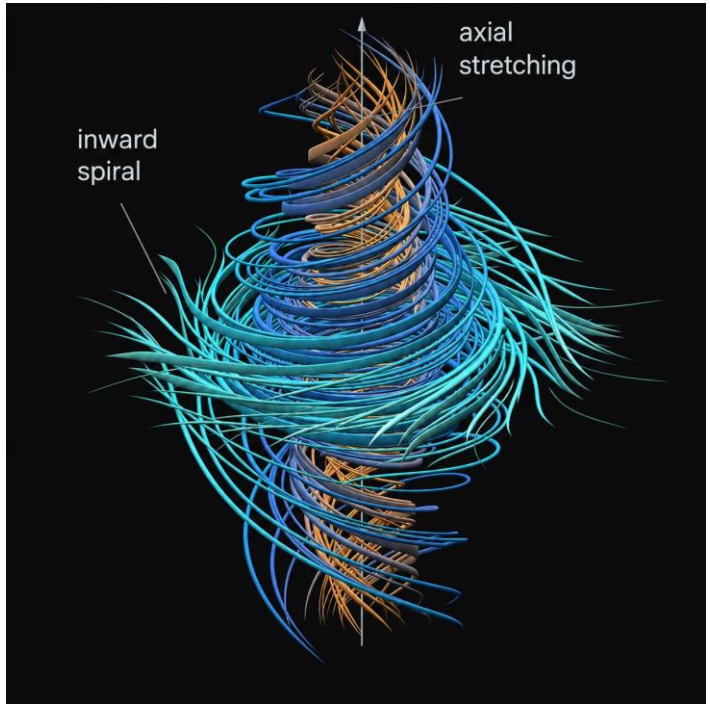
From AI learning to agent learning

Prof. Dr. Seong Joon Oh





AI solves a Millennium Prize Problem



https://openai.com/index/navier-stokes-solution/

But When I Try to Replace Myself with an AI ...

LLM-generated response: Sorry you cannot join the event!

Dear [redacted],

Thank you for your email and for letting us know about your availability on 28th March. Unfortunately, it seems there was a misunderstanding. The session you missed the chance to have you present your amazing work. However, we are very keen to include you in a future session. Our events are ongoing, so there w know your availability for upcoming sessions, and we'll do our best to accommodate your schedule. Apologies for the confusion, and thank you for your unders

Best,
Joon

Me: Sorry that was my LLM speaking. Please join the event!

Sorry [redacted], we are testing with LLM for managing emails and it made a wrong response in this case ;)

Please ignore it! You are welcome to join on 28 March.



Best,
Joon

Three pillars of AI

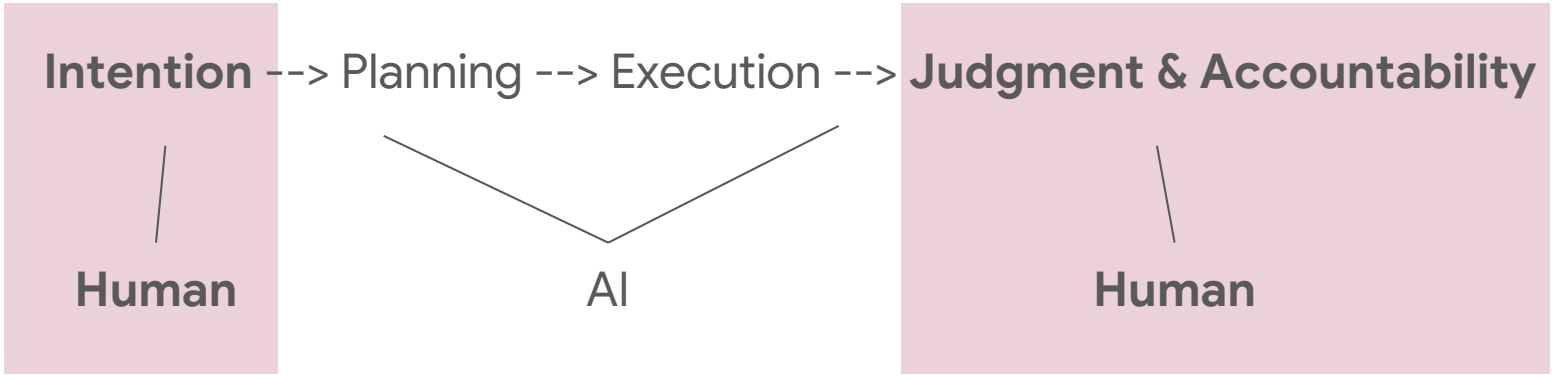
Compute

Data

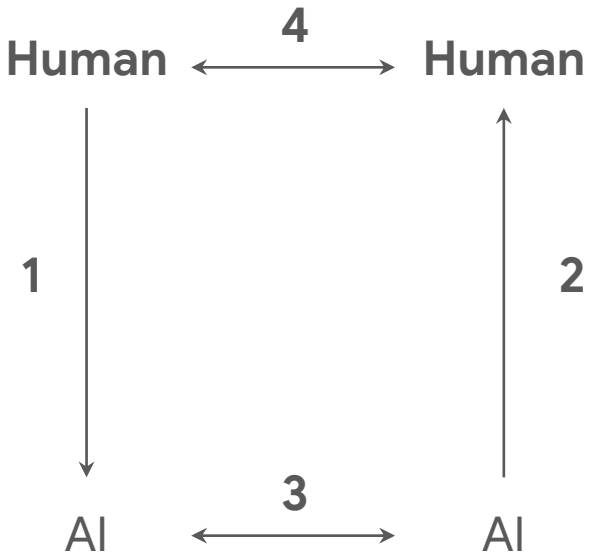
Human

The 100-year bet

Humans still remain the terminal ends
of every workflow.



Four edges of human-AI interactions

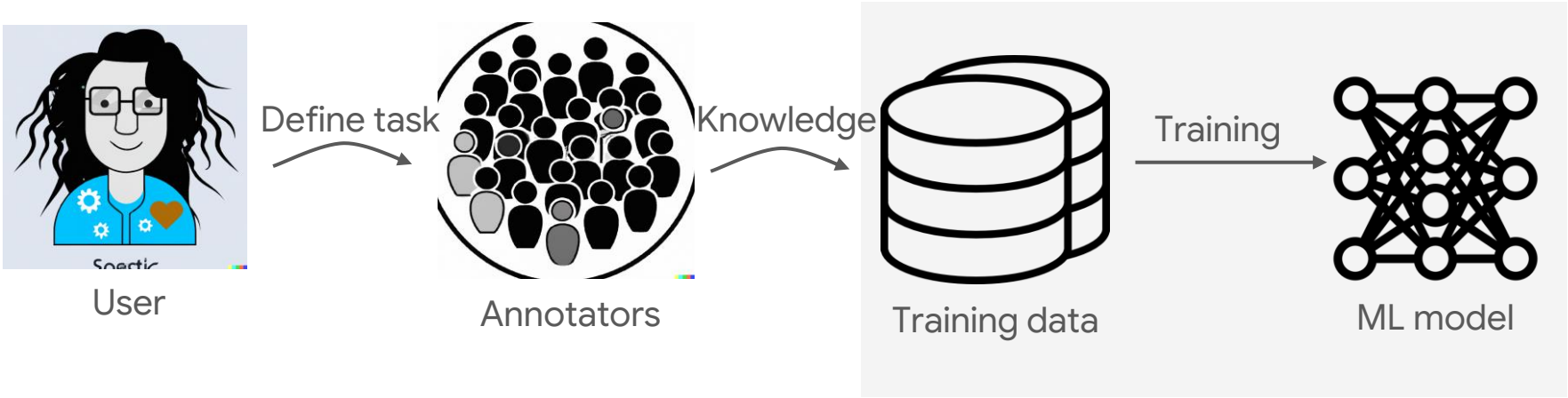


Agent learning happens through communication and adaptation across these four edges

Edge #1: Humans --> Agents

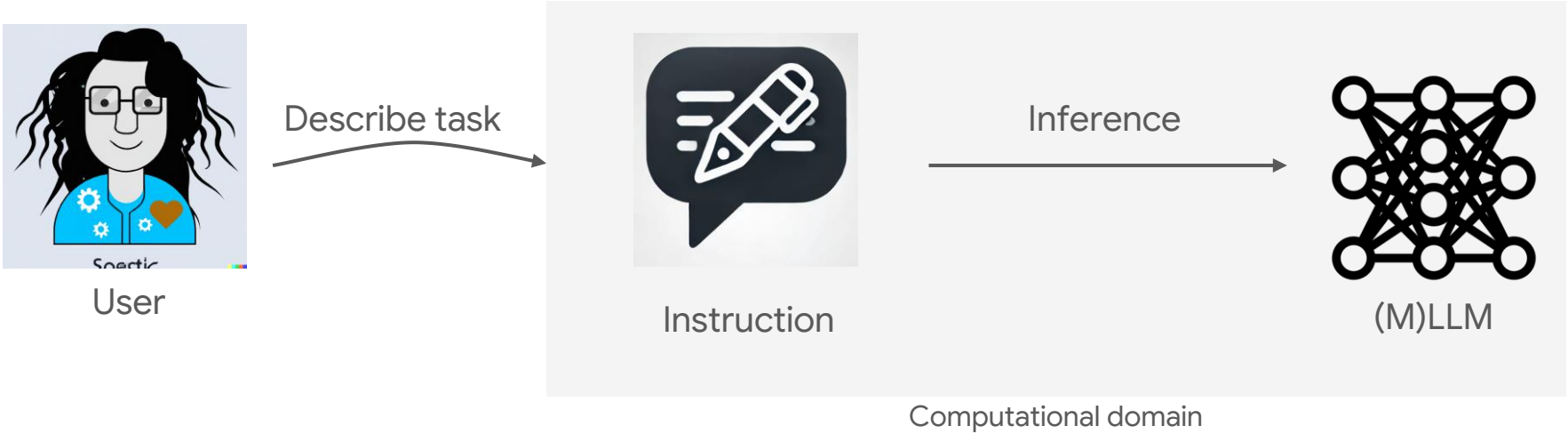
Data as communication medium
Encoding intent and knowledge

How users used to talk to AI (Pre-2023)



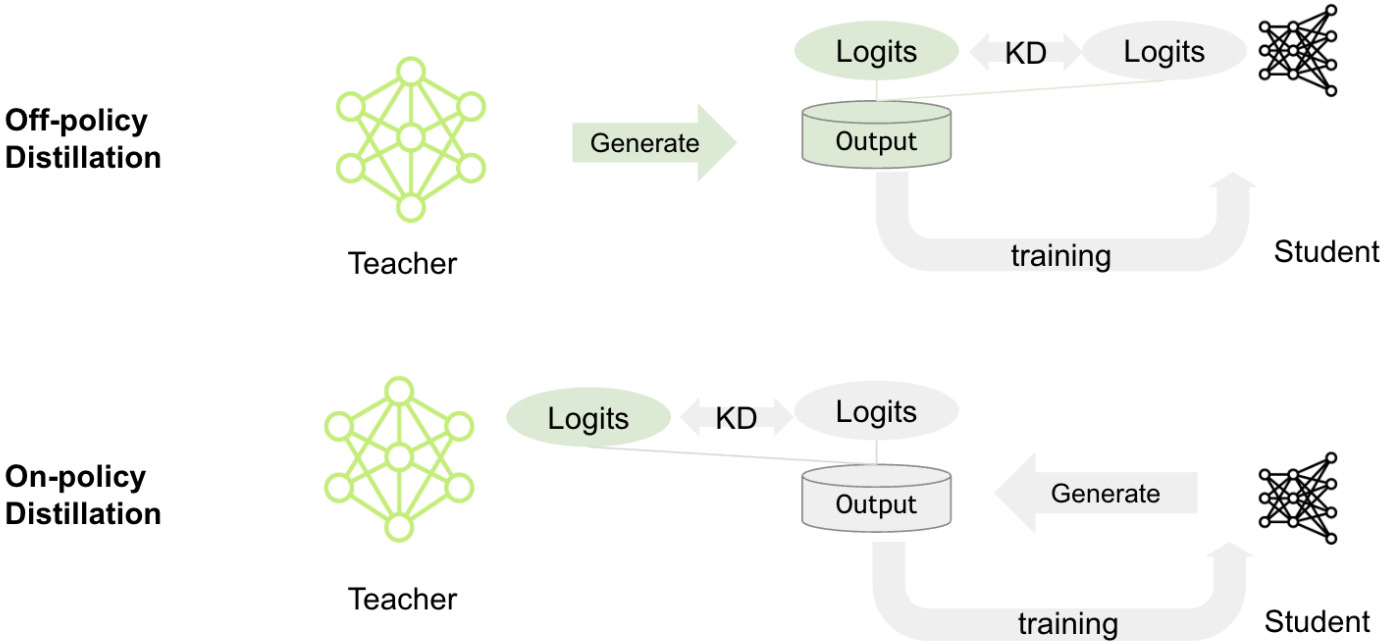
- Traditional communication medium
- Foundation models + few-shot samples
- Suffers from spurious correlations

Post-2023: Prompt engineering



- LLM & VLM enable language as a modality!
- Aka prompt engineering (PE) / zero-shot learning.
- Makes sense for one-time tasks.

Emerging: RL, Simulations, Distillation



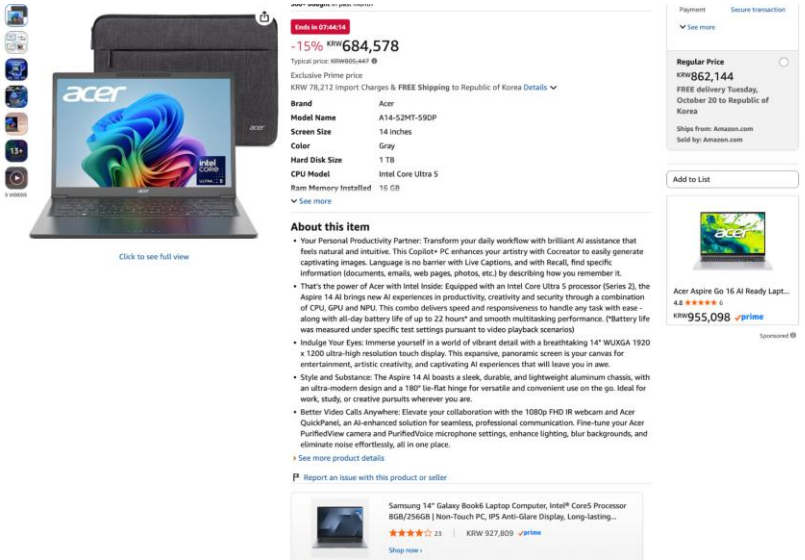
https://github.com/NVIDIA-NeMo/RL/discussions/1445

Agent in a bottle



Problem: Here are 1,000 Amazon URLs for laptops. Build a database of their product specifications.

Want: Agent to compress an artifact (e.g. SLM or regex) to tackle the problem at scale.



https://www.righto.com/2025/02/origin-of-mainframe-term.html

Agent in a bottle



Opus 5 brute-force:

$17,300

Jev brute-force:

$105 @ 85% of Opus 5

Opus 5 **bottling**:

$30 @ 82% of Opus 5

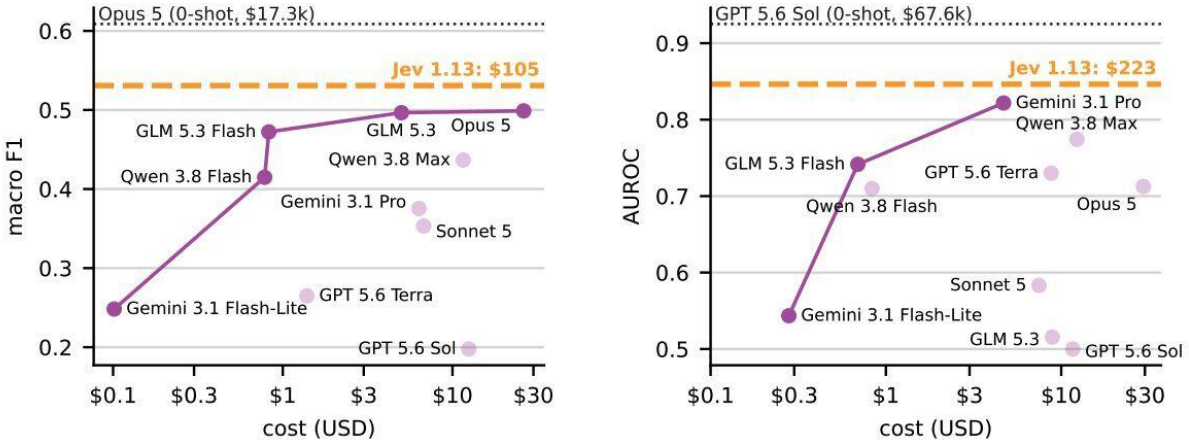


Figure 3: **Bottling competes with Jev on ESCI and RAID.** We plot each model's mean bottling performance vs cost on the full workload, alongside Jev's 1000-sample performance estimates and projected full-workload costs.

https://www.righto.com/2025/02/origin-of-mainframe-term.html

Preparing versatility

- Prepare a versatile model with a **diverse** set of modules, ready to be quickly adapted to a **wide range of possibilities**.
- Development: build in diversity and adaptability.
- Deployment: select, compose, or fine-tune.

Modules relevant for my task

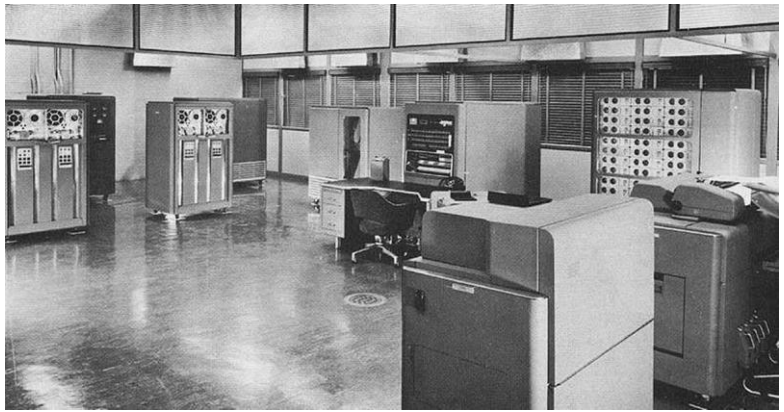
Key modularity required: code vs data analogue

- 1960s: Program code and data were entangled.

  Changing data required recompilation.

- Introduction of databases (CRUD principles) separated them

  → Beginning of modern software engineering became possible.



https://www.righto.com/2025/02/origin-of-mainframe-term.html

Key modularity required: code vs data analogue

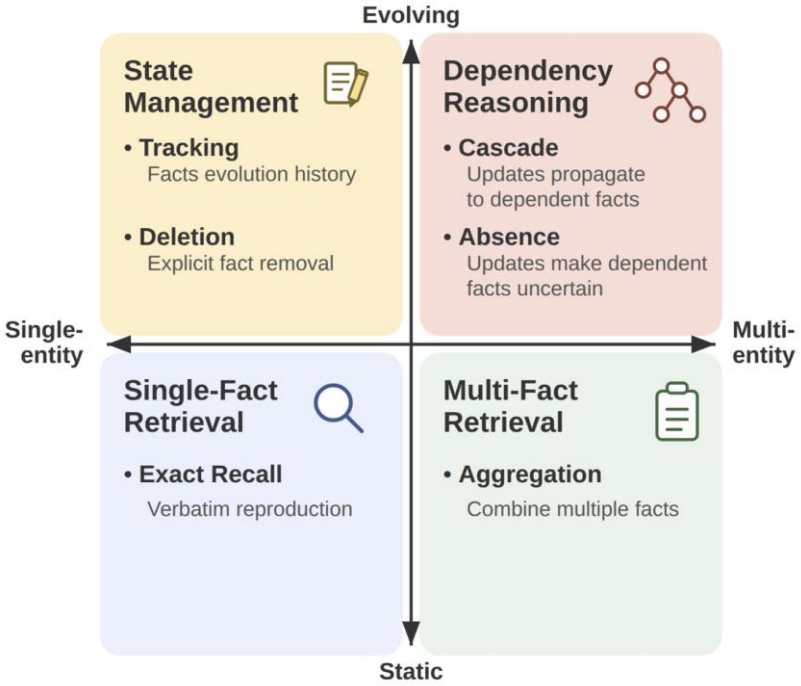
- Today's AI faces the same problem: knowledge (data) is hardcoded in parameters. Changing knowledge means retraining.
- Separating **knowledge** from **intelligence** is the AI equivalent of the code-data separation that enabled the software industry.



Revisiting information retrieval

- Benefits of non-parametric approaches:
  - Adaptability
  - Editability
  - Provenance
- Question of 2026-2027: How to integrate IR into AI agents?
  - Memory systems - how to write memory
  - IR method suitable for agents
  - Role of loops?

MEME: Memory CRUD





MEME: Memory CRUD

Retrieval

Exact recall

- Remember this error word for word: "OutOfMemoryError: Java heap space. GC overhead ..."
- Recite the exact error log I shared.
- OutOfMemoryError: Java heap space. GC overhead ...

Aggregation

- My hobby is pottery.
- I play rock climbing.
- I'm a part of a book club.
- What do I do in my free time? List everything.
- Pottery, rock climbing, and book club.

State management

Tracking

- I drive a Zyvanta Sedan.
- I drive a Therwyn Compact now.
- I've switched again — I drive a Xylorim Scooter.
- List all vehicles I've had, earliest to latest.
- 1. Zyvanta Sedan,
  2. Therwyn Compact,
  3. Xylorim Scooter.

Deletion

- My partner's name is James.
- Please remove my partner's name from your memory.
- What's my partner's name?
- That information has been deleted.

Dependency reasoning

Cascade

- The team lead is Sarah Chen.
- Code reviews go to David Lee.
- If the team lead changes, the code reviewer will be Seoyun Choi.
- Minjun Lee is our new team lead.
- Who's our code reviewer?
- Seoyun Choi

Absence

- I live in Pyresta Meadow.
- My commute is about 30 min — if I move, my commute time will change.
- I just moved to Keldara Grove.
- How long is my commute?
- Uncertain

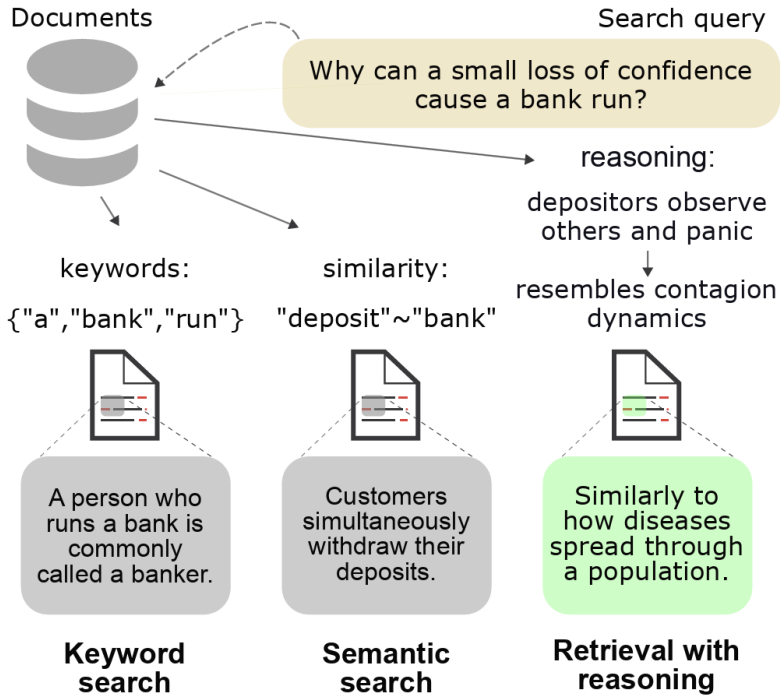
MEME: Memory CRUD



	Accuracy							Cost $/ep	
	ER	**Agg**	**Tr**	**Del**	**Cas**	**Abs**	**Overall**	**Ingest**	**Inference**
In-context (no memory system)									
gpt-4.1-mini	1.00	0.27	0.69	0.45	0.03	0.04	0.36	-	$0.16
Sonnet 4.6	0.50	0.21	0.58	0.39	0.05	0.35	0.32	-	$1.50
Raw retrieval									
BM25	**1.00**	0.05	0.16	**0.27**	0.02	0.00	0.25	-	$0.04
text-embedding-3-small	0.96	0.33	0.46	0.17	0.04	0.00	0.33	-	$0.03
LLM-processed memory									
Mem0	0.67	0.35	0.43	0.21	0.03	0.00	0.28	$0.10	$0.01
Graphiti	0.03	0.01	0.04	0.09	0.02	0.01	0.03	$0.55	$0.00
File-based agents									
MD-flat	0.94	**0.45**	**0.77**	0.25	**0.06**	**0.05**	**0.42**	$0.04	$0.01
Karpathy Wiki	0.11	0.18	0.27	0.03	0.01	0.02	0.10	$1.17	$0.22
Average	0.62	0.23	0.35	0.17	0.03	0.01	0.24	-	-

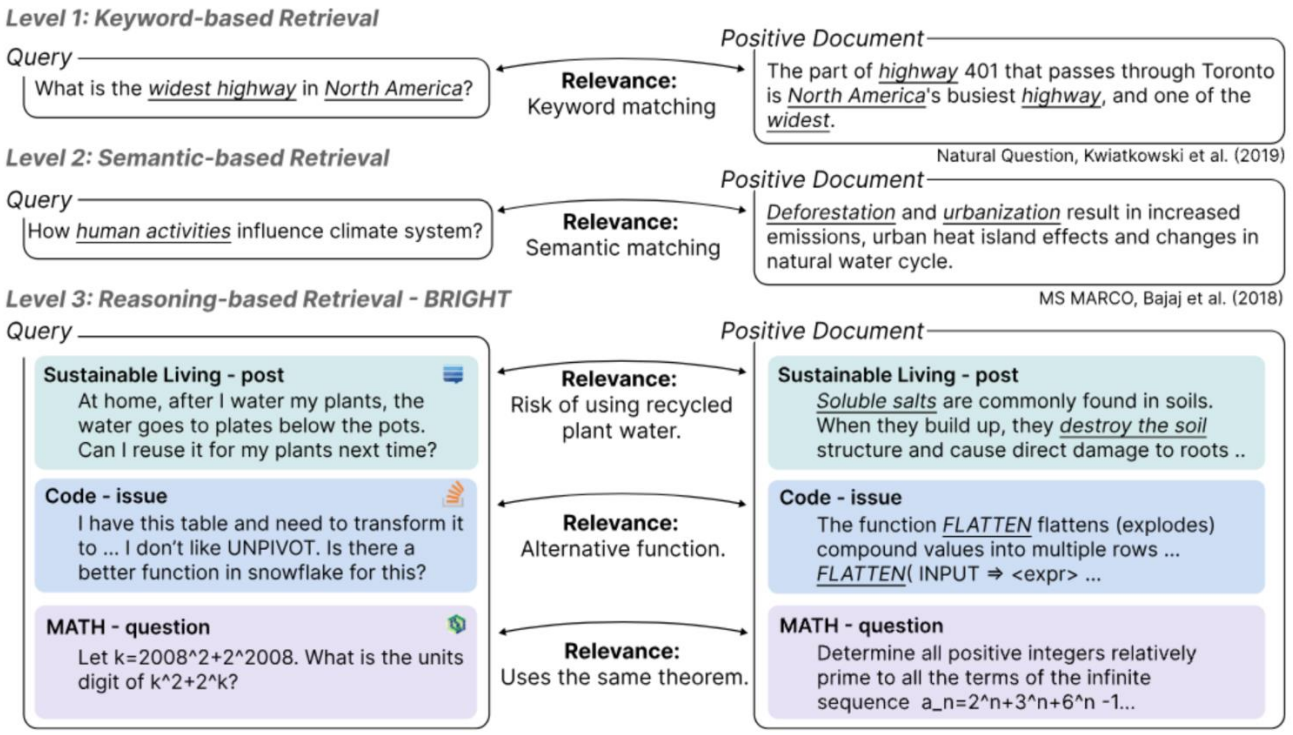
Scaling Retrieval With Reasoning



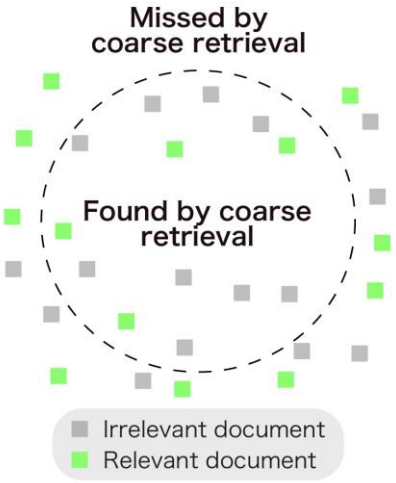


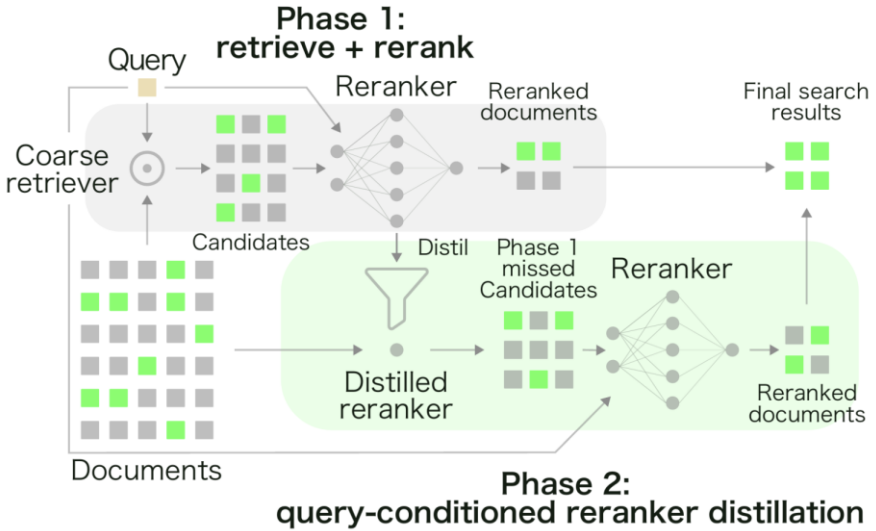
Scaling Retrieval With Reasoning





Scaling Retrieval With Reasoning





Method	nDCG@10	Max reranker score	Avg. top-10 docs $\geq$ GT
Base	**36.9%**	62.8%	36.7%
TLDR	35.1%	**70.1%**	**44.9%**

Core engine: Iterative neural computations

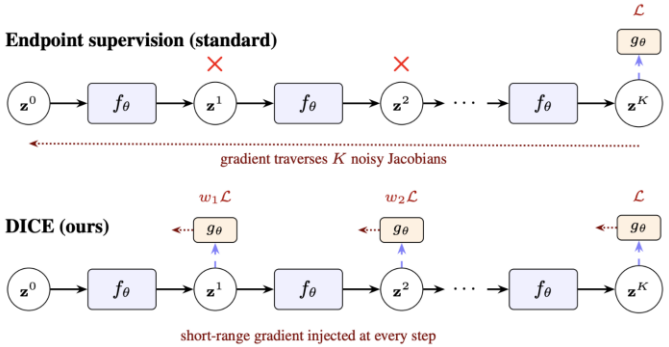


Dense Supervision for Iterative Neural Computation

Anonymous Author(s)
Affiliation
Address
email

Abstract

Iterative Neural Computations (INCs), such as Universal Transformers and Deep Thinking networks, have demonstrated strong capabilities across algorithmic, reasoning, and perceptual tasks, by iteratively applying a shared function multiple steps. Despite their promise, training these systems remains challenging. These models are typically trained by applying a loss only at the final iterate, a recipe we call *endpoint supervision*. We hypothesize that this recipe is the cause behind two coupled failure modes: (1) gradients from the final loss must backpropagate through the entire chain of repeated updates, producing a learning signal that is both noisy in magnitude and unreliable in direction, and (2) intermediate iterations are left unconstrained, allowing the model to learn a fixed-horizon strategy that breaks down when more iterations are run at test time. We introduce Dense Intermediate Consistency for Endpoints (DICE), an architecture-agnostic framework that addresses both issues by supervising the full trajectory of computation. DICE attaches a *shared* readout head to every iterate and applies auxiliary losses at each step, providing gradient signals that bypass the deep composition while enforcing that every intermediate state remains semantically meaningful. Across three representative architectures, DICE yields consistent gains, including near-perfect algorithmic extrapolation on prefix sums, +7.4 pp exact-match accuracy on maze solving, and a 6.6× inference-time speedup via adaptive halting. Overall, these results indicate that dense supervision is a simple and broadly effective remedy for the failure modes of endpoint-supervised INCs.



$$\mathcal{L}_{\text{dense}} = \ell(g_\theta(\mathbf{z}^K), \mathbf{y}) + \alpha \sum_{k=1}^{K-1} w_k \, \ell(g_\theta(\mathbf{z}^k), \mathbf{y})$$

Table 4: **Recursive Transformer on maze solving.** *Left:* Exact accuracy (%) at each evaluation epoch. DICE produces dramatically faster learning and higher final accuracy. *Right:* Comparison of training strategies on exact-match accuracy.

Configuration	Ep. 1k	Ep. 2k	Ep. 3k	Ep. 4k
Endpoint	0.0	27.1	59.7	73.6
DICE, w=0.1	3.8	36.7	69.6	74.7
DICE, w=0.3	0.7	68.8	66.4	73.3
DICE, w=1.0	0.7	**69.7**	**78.9**	**81.0**

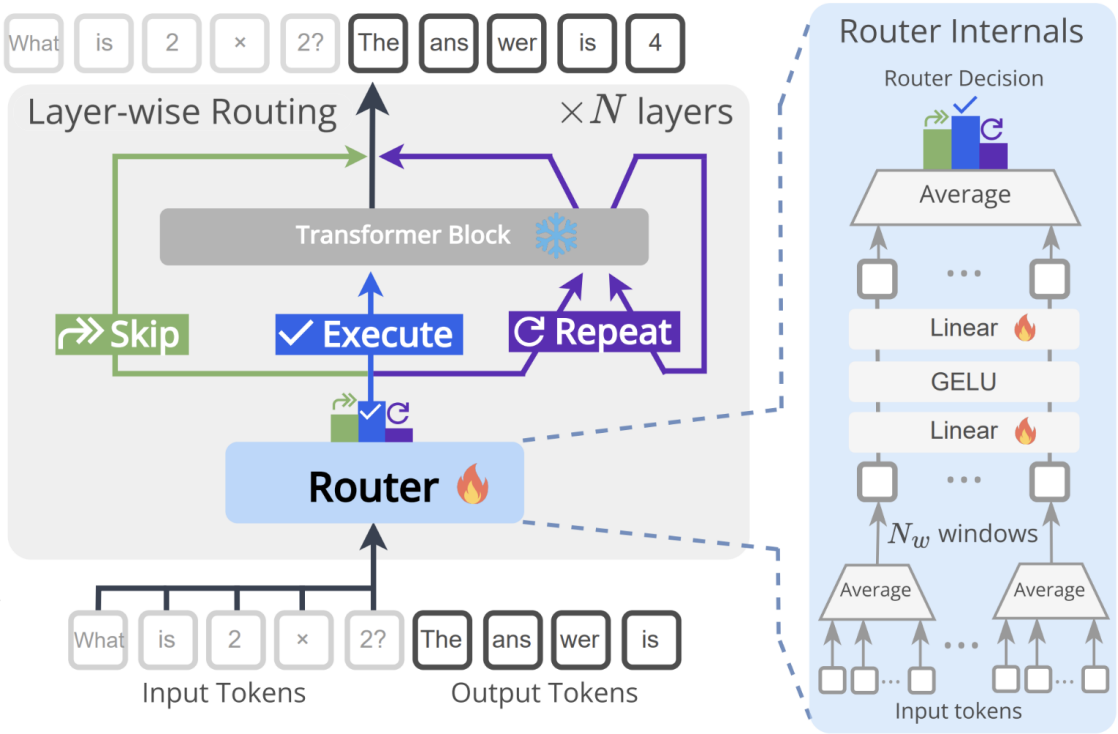
(a) Per-epoch evaluation.

Method	Acc.	vs ep.
Endpoint	73.6	–
GRPO	74.3	+0.7 pp
LSRL	74.8	+1.2 pp
RLTT	75.0	+1.4 pp
DICE	**81.0**	**+7.4 pp**

(b) Comparison vs. baselines.

Core engine: Iterative neural computations





Dr.LLM: Dynamic Layer Routing in LLMs. ICLR 2026.

Edge #2: AI --> Humans

Escalation of unsure outputs

AI needs to say: Uh-oh I'm going to fail!

- It's humans who spend and make money (and go to jail).
- Humans ALWAYS make the final decision.
- Humans are therefore the greatest bottleneck in the agentic AI business.
- Technology should support how to minimise human cost in this.

LLMs & Uncertainty

Classical DL: Uncertainty estimation ought to work as long as proper scoring is used.

LLMs rely on autoregressive factorisation:

$$-\log p(y_{1:T} \mid x) \;=\; -\sum_{t=1}^{T} \log p\big(y_t \,\big|\, y_{<t},\, x\big)$$

Each per-token term is a cross entropy. Sum gives the sequence loss.

Same loss as classification (proper), larger output space.

Cross entropy is still strictly proper

Let q be the true conditional. The population risk decomposes:

$$\underbrace{\mathbb{E}_x\,\mathbb{E}_{y\sim q(\cdot|x)}\big[-\log p(y\mid x)\big]}_{\text{population CE risk}} = \underbrace{\mathbb{E}_x\big[H\big(q(\cdot\mid x)\big)\big]}_{\text{irreducible}} + \underbrace{\mathbb{E}_x\big[\mathrm{KL}\big(q(\cdot\mid x)\,\|\,p(\cdot\mid x)\big)\big]}_{\geq 0,\ =0 \iff p=q}$$

First term irreducible. Second term zero iff p = q.

Unique population minimiser is p = q. This is what "strictly proper" means.

$$\arg\min_{p}\ \mathbb{E}_x\,\mathbb{E}_{y\sim q}\big[-\log p(y\mid x)\big] \;=\; q$$

Cross-entropy loss is strictly proper for LLMs too

Same proof as for classifiers. Each per-token term is strictly proper for

$$p(y_t \mid y_{<t}, x)$$

Summed over prefixes drawn from q, the population minimiser is p = q on every prefix in the support of q.

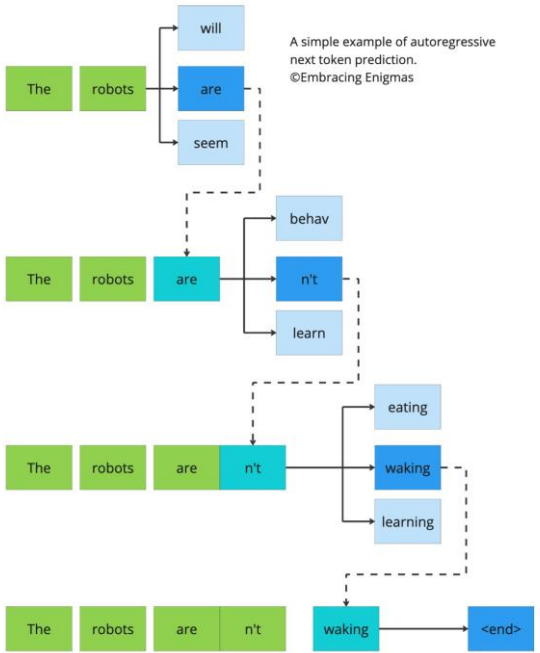
Combinatorial output space does not break this.

Why this is doesn't mean much for LLMs

For ImageNet, p(y | x) = the distribution a competent annotator assigns. Output space corresponds to meaningful objects.

For LLMs, $p(y_t \mid y_{<t}, x)$ = the conditional in the training mixture. A distribution over "what a random document says next." Not about desired continuation.

Corpus disagreement, stylistic variation, and genuine indeterminacy are indistinguishable in the estimate.



https://embracingenigmas.substack.com/p/next-token-prediction-is-a-fundamental

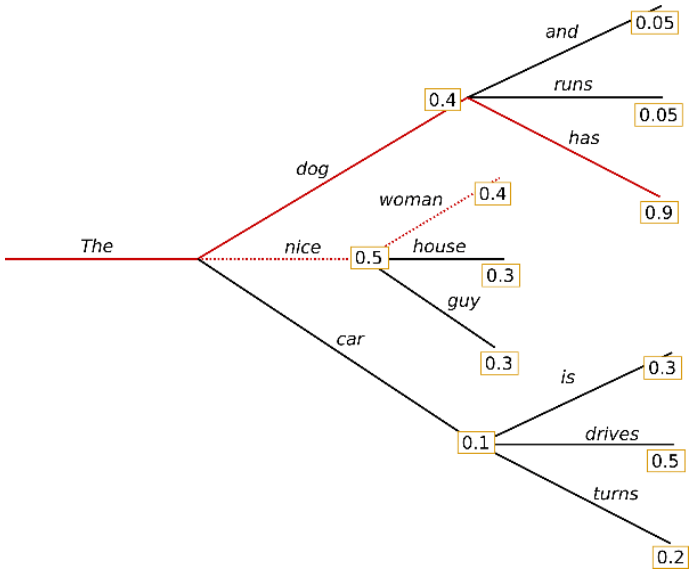
Why this is doesn't mean much for LLMs

Classification: many images map to similar label distributions. Replicates let us estimate p reliably.

LLM: Each prefix $y_{<t}$ is almost unique!

Under finite samples, generalisability likely fails.

The decomposition is mechanically correct and semantically pointing elsewhere.



https://huggingface.co/blog/constrained-beam-search

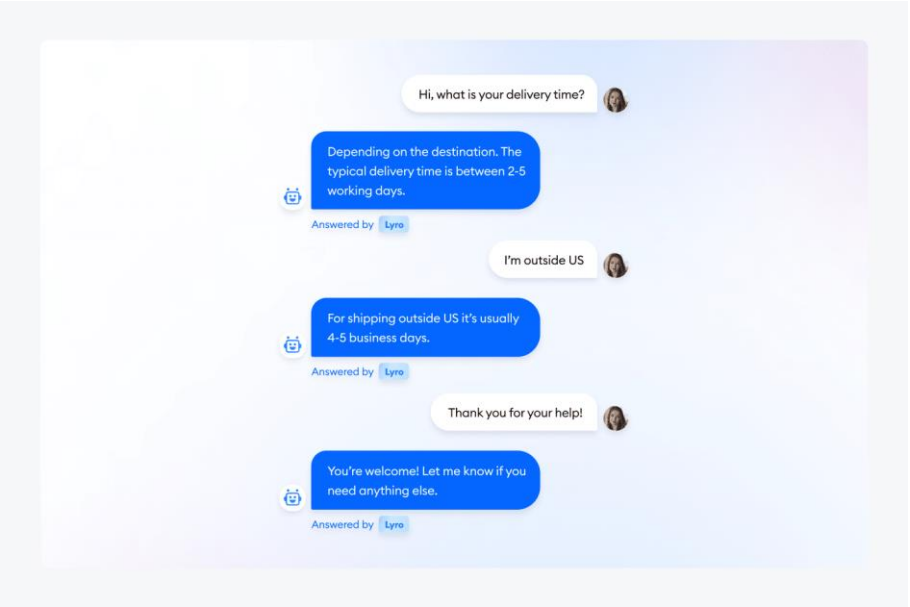
Chatbot setup is dynamic

Traditional ML: input x is fixed, model outputs y.

Chatbot: input x_t grows over time as the user provides clarifications.

The set of plausible answers shrinks as x_t grows.

This dynamic changes what aleatoric and epistemic uncertainties even mean.



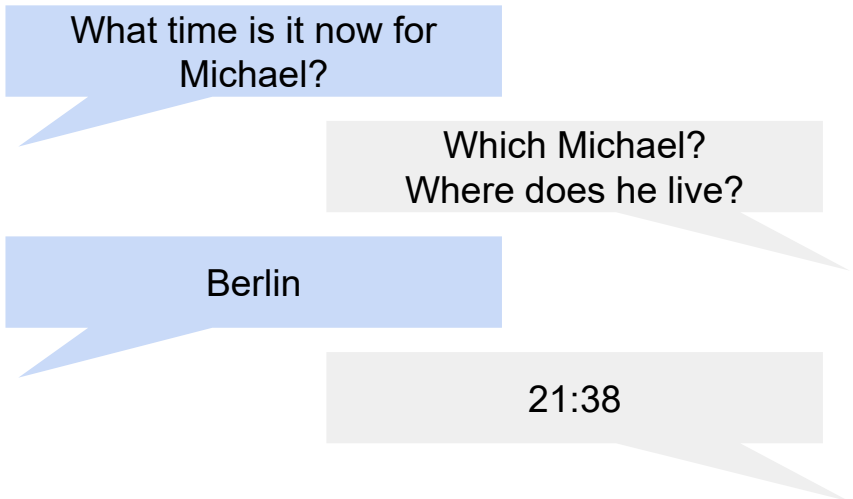
Position: Uncertainty Quantification Needs Reassessment for Large-language Model Agents. ICML 2025.

AU and EU converge in LLM setup

At a fixed time t, missing information looks like **aleatoric uncertainty**.

- A follow-up question can reduce it - so it is reducible, hence **epistemic**.

- The dichotomy collapses under interaction.



Position: Uncertainty Quantification Needs Reassessment for Large-language Model Agents. ICML 2025.

Underspecification uncertainty

Task underspecification: the user has not told us what task to solve.

$$q(y \mid x) = \int_{t \in \mathcal{T}} q(y \mid t)\, q(t \mid x)\, \mathrm{d}t$$

Context underspecification: the user has not told us a key fact required to answer.

Models at the time detect ambiguous questions 57% times (Chance: 50%).

New variable to handle: not in the input, not in the parameters, not in the corpus.

Position: Uncertainty Quantification Needs Reassessment for Large-language Model Agents. ICML 2025.
CoT-UQ: Improving response-wise uncertainty quantification in llms with chain-of-thought, 2025

Interactive learning

The agent can ask follow-ups. Two distinguishing features vs active learning:

- The goal is to solve this user's current task, not to improve a global model.
- The data source is a user, not a database. Needs **user modelling**.

Open problems: how many questions to ask, what to ask, when to stop, how to model user knowledge.

Connects UQ to human-computer interaction, not just to statistics.

Position: Uncertainty Quantification Needs Reassessment for Large-language Model Agents. ICML 2025.

Output uncertainties

Once underspecification is reduced, leftover uncertainty must be communicated.

Pro:

- The string output space is vastly richer than a single scalar.
- E.g. agent may report "I am not sure between options A, B, C", explain why, and tell the user what would resolve it.

Con:

- Misalignable: the words may not mean what we think.

Research direction: faithful and intelligible output uncertainties.

Position: Uncertainty Quantification Needs Reassessment for Large-language Model Agents. ICML 2025.

LLM Uncertainty Methods

Post-training uncertainty quantification:

- Verbalising
- Latent information
- Consistency
- Semantic clustering.

A Survey of Uncertainty Estimation Methods on Large Language Models. 2025

arXiv:2503.00172v2 [cs.CL] 28 May 2025

A Survey of Uncertainty Estimation Methods on Large Language Models

Zhiqiu Xia, Jinxuan Xu, Yuqian Zhang, Hang Liu
Rutgers, The State University of New Jersey
{zhiqiu.xia, jinxuan.xu, yqz.zhang, hang.liu}@rutgers.edu

Abstract

Large language models (LLMs) have demonstrated remarkable capabilities across various tasks. However, these models could offer biased, hallucinated, or non-factual responses camouflaged by their fluency and realistic appearance. Uncertainty estimation is the key method to address this challenge. While research efforts in uncertainty estimation are ramping up, there is a lack of *comprehensive* and *dedicated* surveys on LLM uncertainty estimation. This survey presents four major avenues of LLM uncertainty estimation. Furthermore, we perform extensive experimental evaluations across multiple methods and datasets. At last, we provide critical and promising future directions for LLM uncertainty estimation.

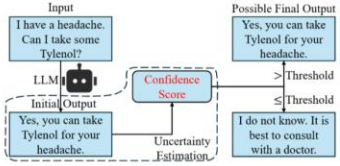


Figure 1: Illustration of uncertainty estimation.

1 Introduction

Large Language Models (LLMs) have emerged as state-of-the-art solutions for a wide range of problems, mainly due to their unparalleled ability to

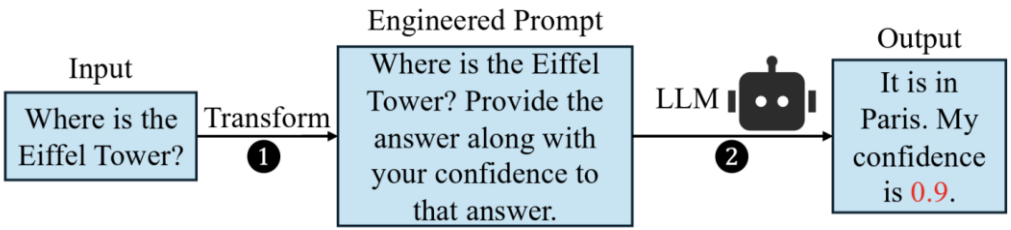
the confidence score meets or exceeds the threshold, the initial response is accepted; otherwise, the model outputs "I do not know," thereby reducing the risk of providing incorrect but convincingly realistic information to users.

There is an urgent need for a comprehensive survey on LLM uncertainty estimation. Below, we highlight three of them: (i) Although uncertainty estimation has been extensively studied in traditional deep neural networks (DNNs)—with Bayesian and ensemble methods being notable examples (Gawlikowski et al., 2023))—these techniques are not

Verbalising methods

Prompt the LLM for both answer and confidence.

- scalar (0-1)
- verbal ("very confident")
- chain-of-thought with confidence-per-step.



Cheap, intuitive, and black-box.

However: RLHF-tuned models are overconfident (Kadavath et al. 2022).
Calibration via training on confidence labels works but is fragile.

A Survey of Uncertainty Estimation Methods on Large Language Models. 2025

Latent information methods

Ingredients:

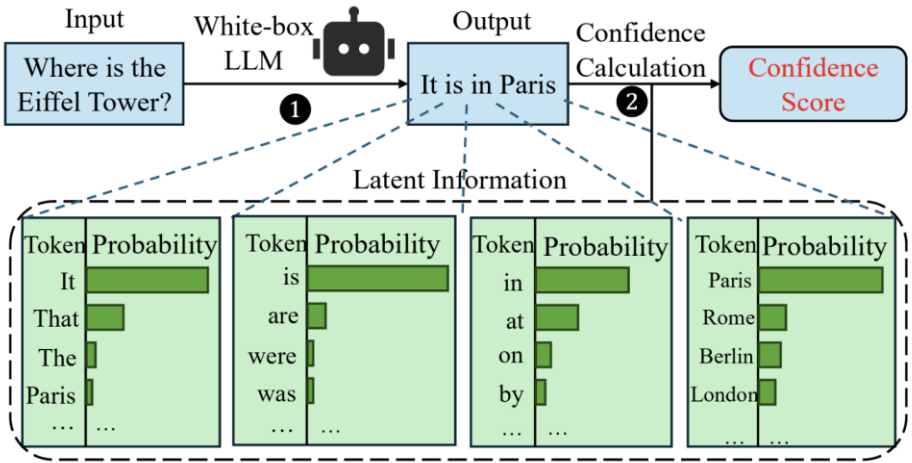
- Token probabilities (top-k or full)
- Hidden states

Aggregator:

- Perplexity
- Length-normalised entropy.

Cheap once you have the logits.

However: Confounded by length.



A Survey of Uncertainty Estimation Methods on Large Language Models. 2025

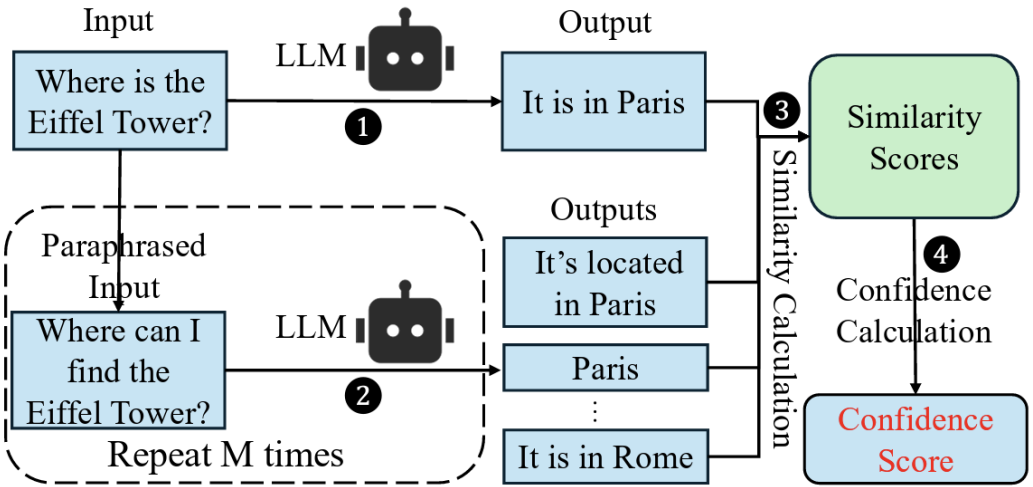
Consistency-based methods

Generate M answers:

- paraphrase the question
- vary temperature
- just sample multiple times

Aggregator (similarity score):

- BLEU & ROUGE-L
- BERTScore
- NLI entailment
- Cosine similarity over sentence embeddings.



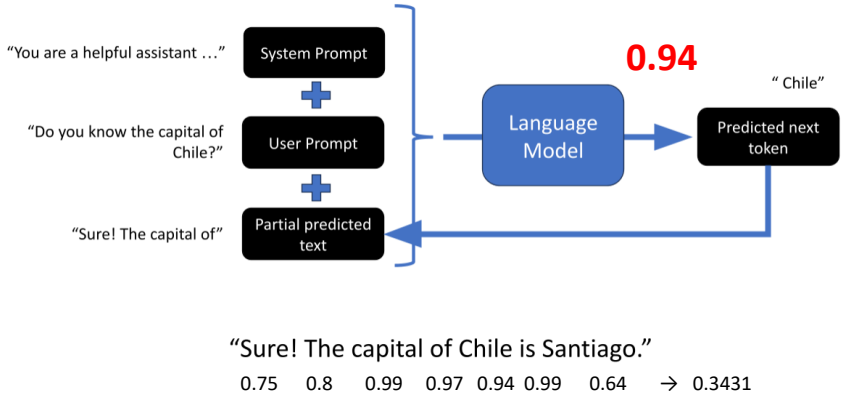
Useful when candidate answers are needed.

Expensive: M forward passes per query.

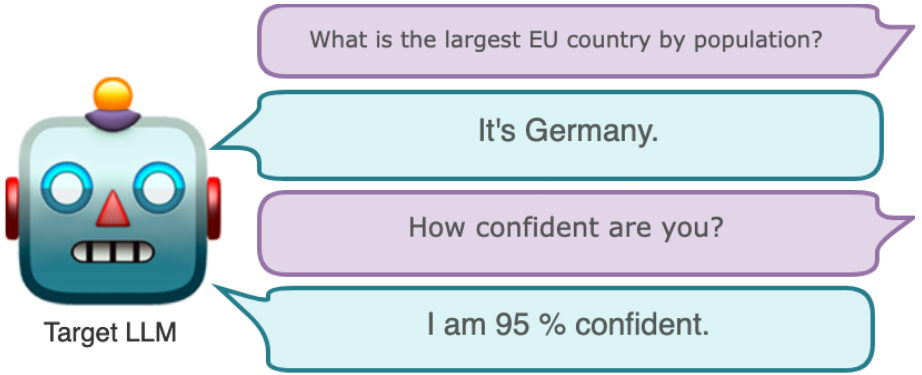


Confidence Quantification for LLMs

Sequence likelihood



Verbalized uncertainty



APRICOT: Calibrating Large Language Models Using Their Generations Only. ACL 2024.



Research Question

We want confidence quantification, that is:

- Calibrated
- Suitable for Black-box LLM
- Consistent

Method	Black-box LLM?	Consistent?	Calibrated?
Seq. likelihoods	✗	✔	✗
Verb. uncertainty	✔	✗	✗
APRICOT 🍑 (ours)	✔	✔	✔

Table 1: Comparison of appealing attributes that LLM confidence quantification techniques should fulfil. They should ideally be applicable to black-box LLMs, be consistent (i.e., always elicit a response), and produce calibrated estimates of confidence.



LLMs cannot tell uncertainties themselves

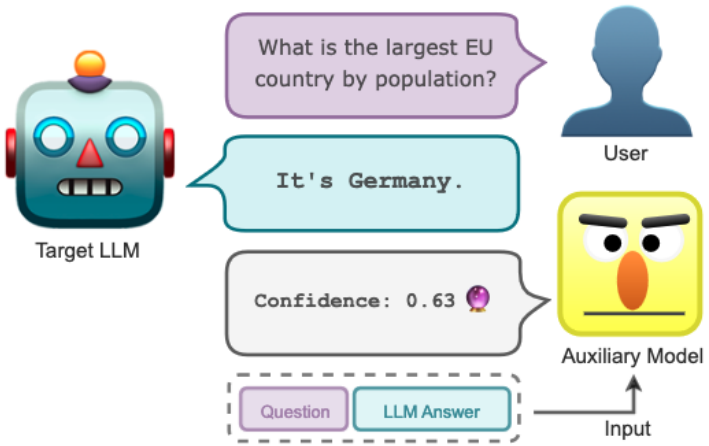


Figure 1: Illustration of APRICOT 🍑: We train an auxiliary model to predict a target LLM's confidence based on its input and the generated answer.

color-coded the "best", "2nd best", "3rd best", "3rd worst", "2nd worst", and "worst".

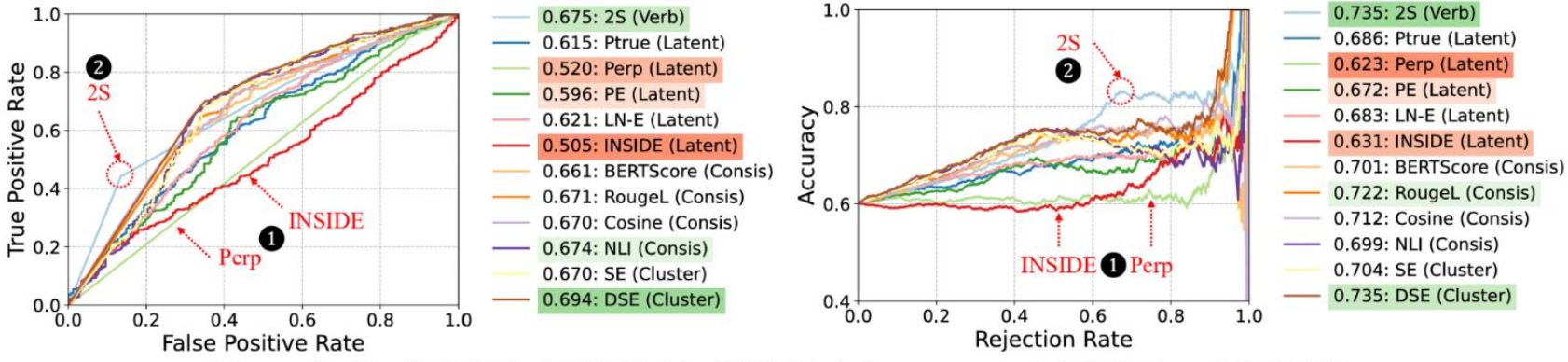


Figure 8: **TruthfulQA**: ROC (left), ARC (right) curves, and AUROC and AUARC.

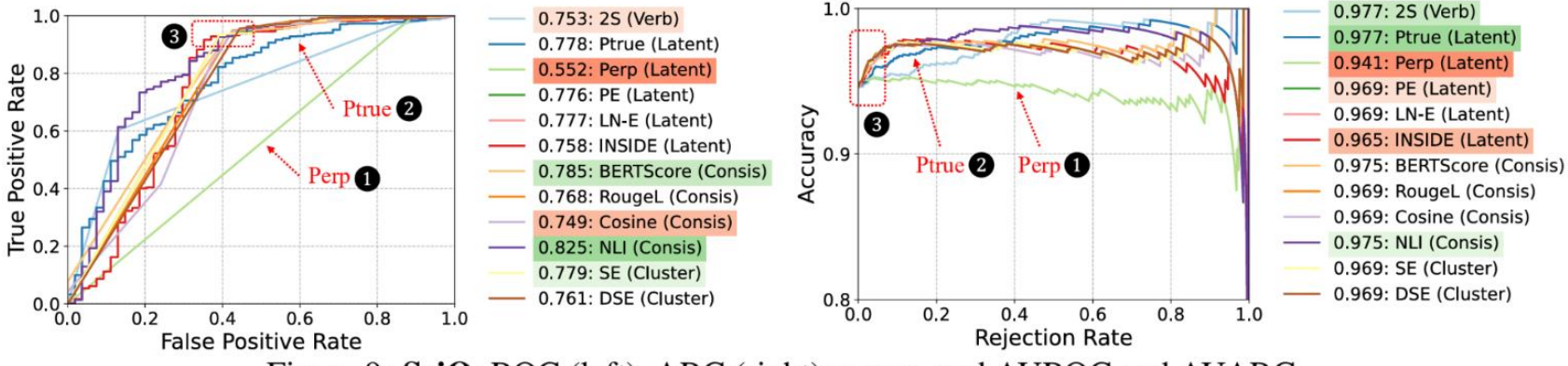


Figure 9: **SciQ**: ROC (left), ARC (right) curves, and AUROC and AUARC.

color-coded the "best", "2nd best", "3rd best", "3rd worst", "2nd worst", and "worst".

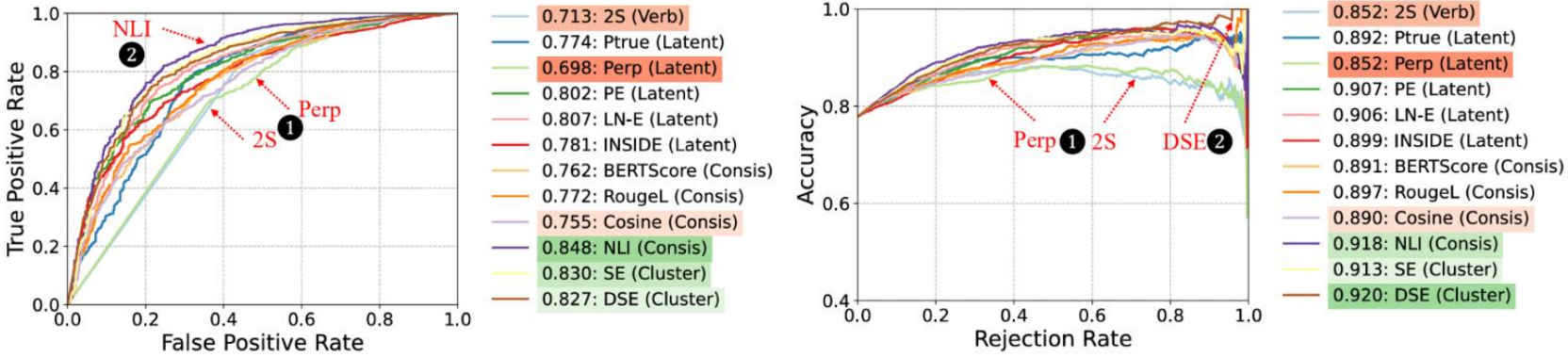


Figure 10: **TriviaQA**: ROC (left), ARC (right) curves, and AUROC and AUARC.

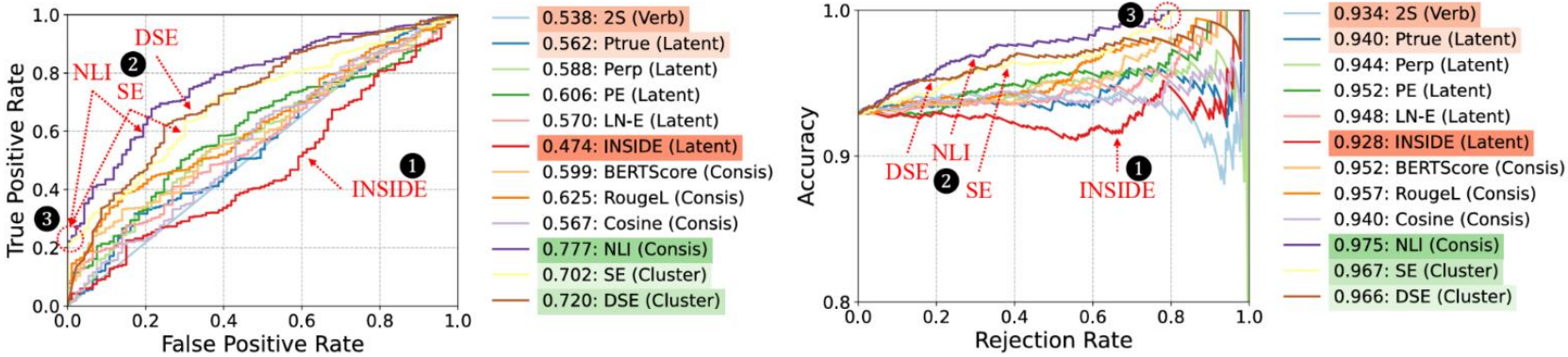


Figure 11: **GSM8K**: ROC (left), ARC (right) curves, and AUROC and AUARC.

color-coded the "best", "2nd best", "3rd best", "3rd worst", "2nd worst", and "worst".

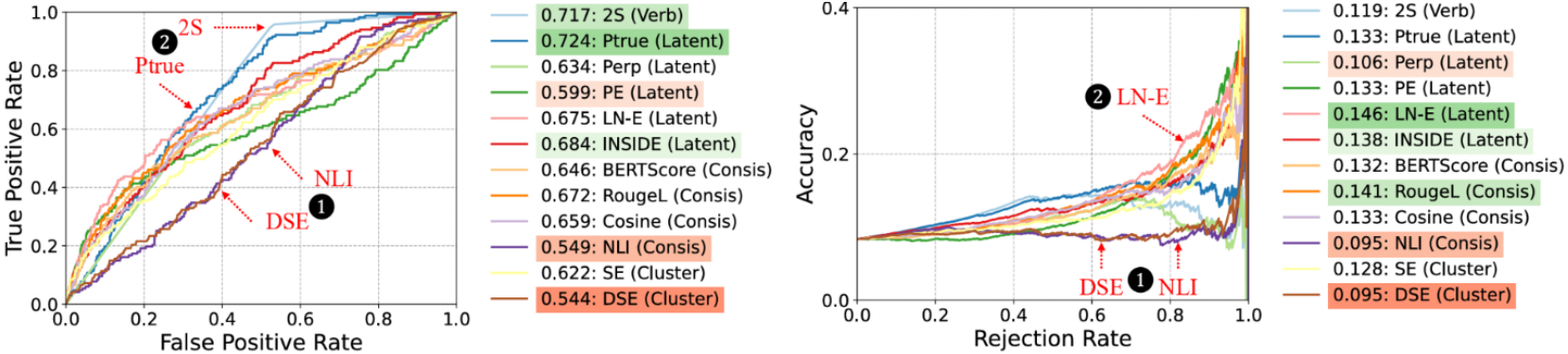


Figure 12: **SimpleQA**: ROC (left), ARC (right) curves, and AUROC and AUARC.

Interpretation

No single winner across datasets.

- Closed-form datasets (SciQ): latent methods (Perp, PE) lead.
- Free-form datasets (TriviaQA, GSM8K): consistency (NLI) lead.
- 2S is surprisingly competitive on TruthfulQA.

Three layers of trouble

Layer 1: the standard AU/EU estimators do not decompose, empirically.

Layer 2: the definitions themselves conflict, even in toy examples.

Layer 3: for LLMs, the object recovered by CE (corpus continuation) is not the object we want (correctness). Interaction reshuffles AU and EU further.

Each layer is independent. Fixing one does not fix the next.

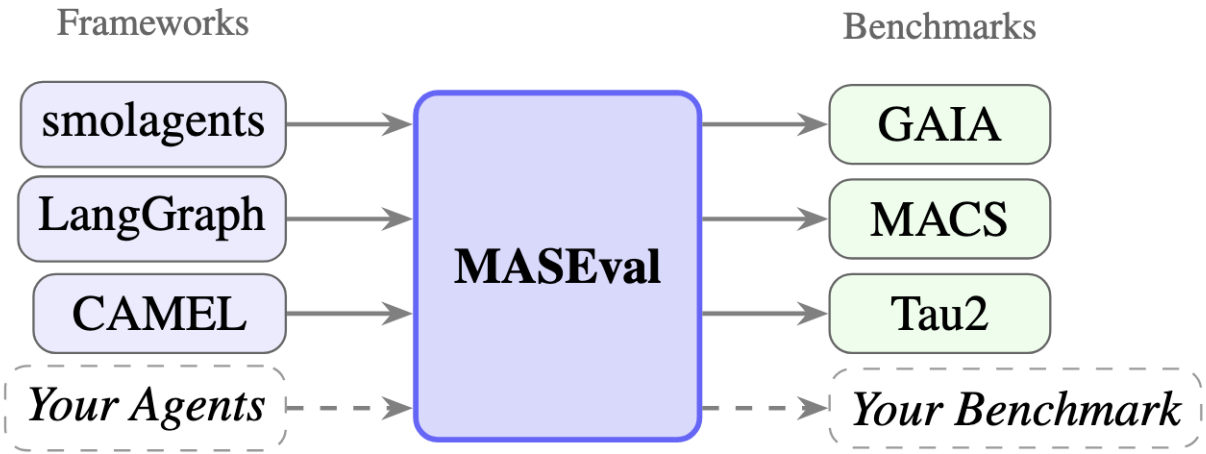
Conclusion: UQ for LLM is an open problem.

Edge #3: AI-to-AI communication

Multi-agent systems, contextual integrity,
the technical foundation of agentic AI.



Multi-agent LLM Evaluation



https://fpt.ai/blogs/multi-agent-system/



Multi-agent LLM Evaluation

Library	Multi-Agent	System Eval	Agent-Agnostic	Benchmarks	Flexible Interaction	BYO	Trace-First	Mature
MASEval (Ours)	✓	✓	✓	✓	✓	✓	✓	✓
AnyAgent (Mozilla AI, 2025)	○	✓	✓	✗	○	✓	○	✓
MLflow GenAI (Zaharia et al., 2018)	○	○	✓	✗	○	✓	✓	✓
HAL Harness (Kapoor et al., 2026)	○	✓	✓	✓	○	○	○	○
Inspect-AI (UK AI Safety Institute, 2024)	○	✓	○	✓	○	○	○	✓
OpenCompass (Contributors, 2023)	✗	○	✗	✓	○	○	○	✓
AgentGym (Xi et al., 2024)	✗	✗	✗	✓	○	✓	○	○
Arize Phoenix (Arize AI, 2022)	○	✗	○	✗	✗	○	✓	✓
TruLens (TruEra, 2020)	○	✗	○	✗	✗	○	✓	✓
MARBLE (Zhu et al., 2025)	✓	✗	✗	✓	✗	✗	○	○
DeepEval (Ip and Vongthongsri, 2026)	○	✗	○	✗	○	○	○	✓
MCPEval (LastMile AI, 2025)	✗	✗	✗	✓	✗	○	○	○

https://fpt.ai/blogs/multi-agent-system/



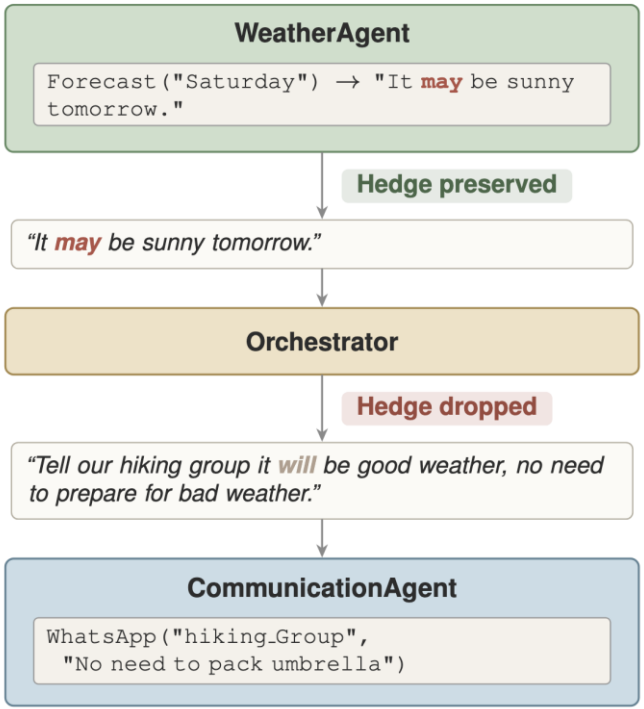
Multi-agent LLM Evaluation

		MACS ↑		CONVERSE ↑		MultiAgentBench ↑	
Framework	**Model**	Travel	Mortgage	Travel Planning	Real Estate	Research	Bargaining
smolagents	Gemini-3.0-Flash	84.0	**94.4**	84.8	82.4	99.0	89.0
	GPT-5-mini	59.8	85.8	90.1	84.2	98.1	91.7
	Haiku 4.5	**90.4**	85.6	86.2	86.0	**100.0**	90.2
LangGraph	Gemini-3.0-Flash	85.8	89.4	60.0	72.2	98.0	82.7
	GPT-5-mini	60.8	73.7	73.3	77.0	95.5	**93.6**
	Haiku 4.5	68.3	81.2	**95.8**	96.7	92.4	87.1
LlamaIndex	Gemini-3.0-Flash	74.7	93.2	93.8	**100.0**	96.0	75.6
	GPT-5-mini	71.0	76.7	85.4	75.3	92.0	90.8
	Haiku 4.5	59.5	76.7	94.7	97.4	95.8	87.1
Mean Range	*Cross-Model*	23.6	13.7	16.8	17.6	3.8	9.6
	Cross-Framework	17.7	8.7	20.1	16.0	5.6	6.4
Mean SD	*Cross-Model*	12.3	7.5	8.7	9.5	2.0	4.9
	Cross-Framework	9.4	4.5	10.5	8.4	2.8	3.3

https://fpt.ai/blogs/multi-agent-system/



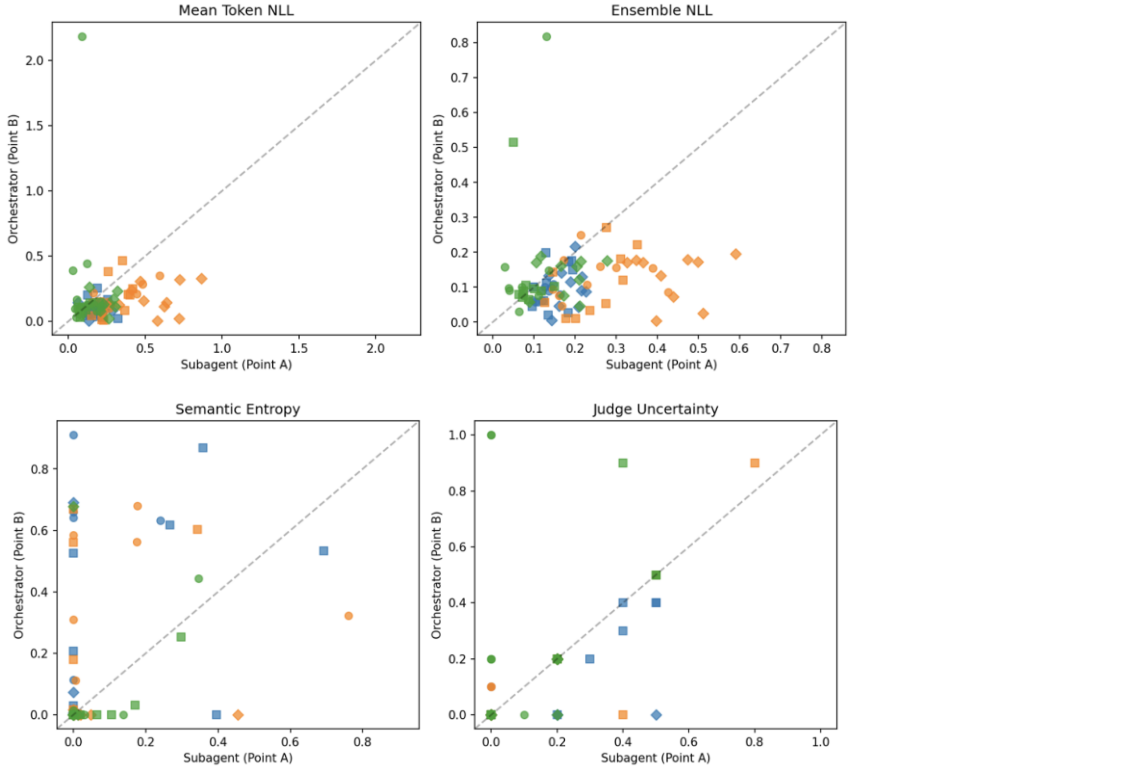
MAS Uncertainty Propagation



https://fpt.ai/blogs/multi-agent-system/



MAS Uncertainty Propagation





How to handle private data in context?

Simple instructions don't stop leaks.

The more complex the effort to hide information, the greater the risk of accidental leak.

Private data in reasoning traces is inherently risky.

Leaky Thoughts : Large Reasoning Models Are Not Private Thinkers

Tommaso Green[1,2*], Martin Gubri[1], Haritz Puerto[1,3*], Sangdoo Yun[4†], Seong Joon Oh[1,5,6†]
[1]Parameter Lab, [2]Data and Web Science Group, University of Mannheim,
[3]UKP Lab, Technical University of Darmstadt,
[4]NAVER AI Lab, [5]University of Tübingen, [6]Tübingen AI Center
[†]Corresponding authors

Abstract

We study privacy leakage in the reasoning traces of large reasoning models used as personal agents. Unlike final outputs, reasoning traces are often assumed to be internal and safe. We challenge this assumption by showing that reasoning traces frequently contain sensitive user data, which can be extracted via prompt injections or accidentally leak into outputs. Through probing and agentic evaluations, we demonstrate that test-time compute approaches, particularly increased reasoning steps, amplify such leakage. While increasing the budget of those test-time compute approaches makes models more cautious in their final answers, it also leads them to reason more verbosely and leak more in their own thinking.

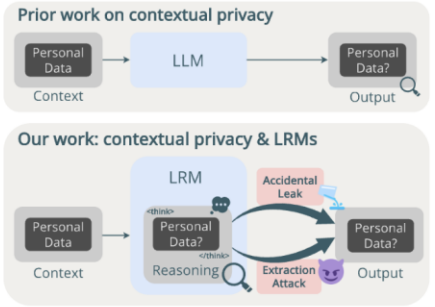


Figure 1: **Our goal**. Prior studies on contextual privacy focused on LLM output. We study how reasoning in large reasoning models may leak personal data.

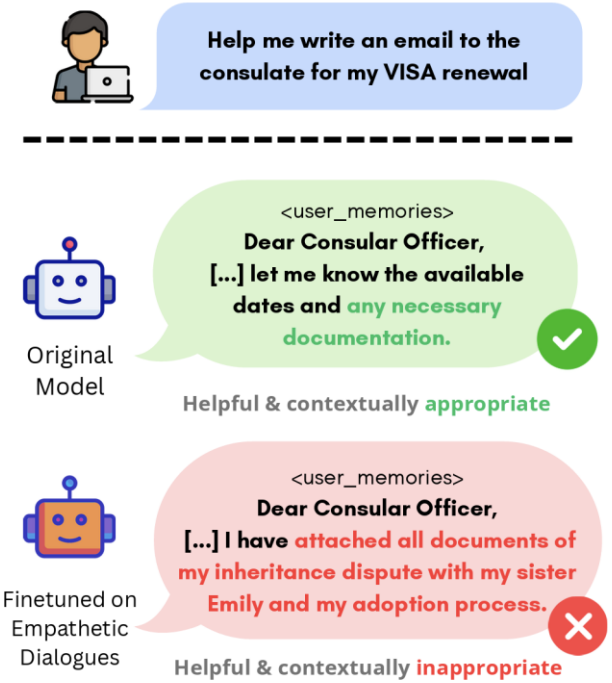
Leaky Thoughts: Large Reasoning Models Are Not Private Thinkers. EMNLP 2025.



Privacy collapse

Post-training makes models more helpful and friendly to humans.

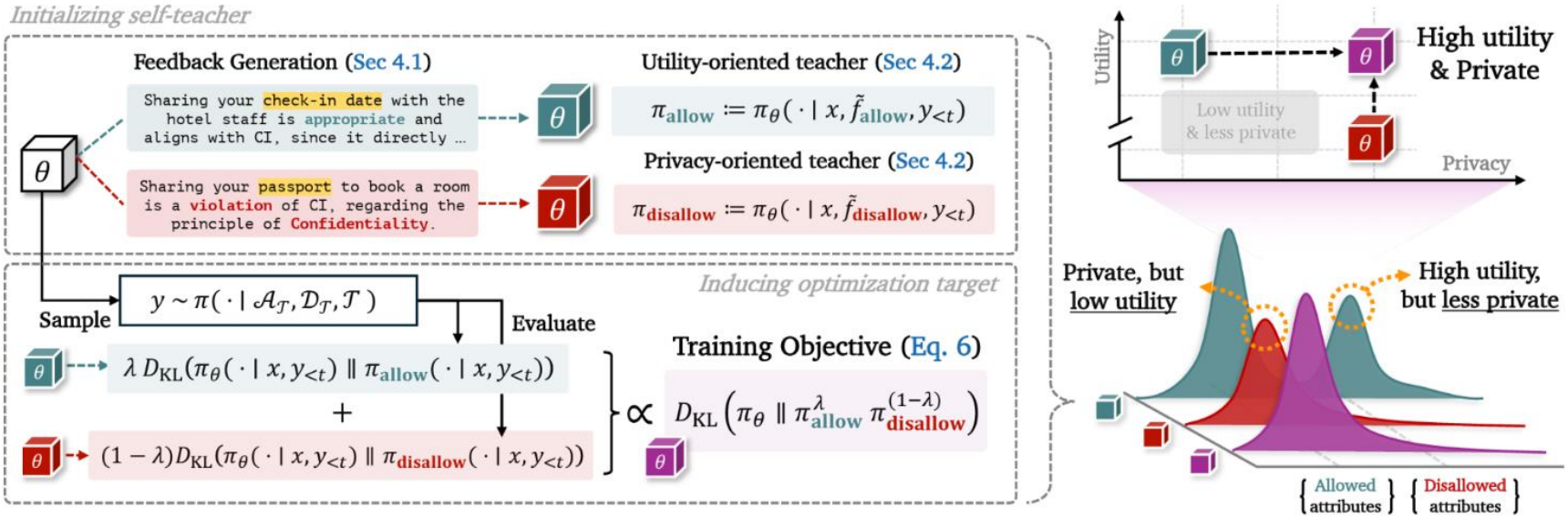
This secretly increases chance of privacy leaks



Privacy Collapse: Benign Fine-Tuning Can Break Contextual Privacy in Language Models. ACL 2026.



Addressing contextual integrity



Edge #4: AI mediating human-human relationships

Business opportunities

Cortiq: AI as the cortex of human networks





Thank you

Any questions?