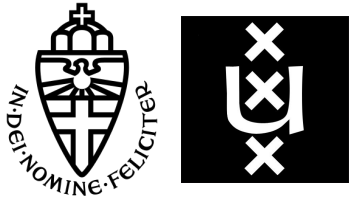


ECIR 2026

Join the Q&A/polls on
ecir2026.eu

ECIR 2026 Tutorial

Reasoning for IR & IR for Reasoning

Mohanna Hoveyda, Panagiotis Eustratiadis, Arjen de Vries, Maarten de Rijke

https://reasoning-for-ir.github.io/

March 29, 2026
Delft, The Netherlands

Presenters

Mohanna Hoveyda

Radboud University

PhD student

Panagiotis Eustratiadis

University of Amsterdam
Bloomberg

Postdoc

Arjen P. de Vries

Radboud University

Professor

Maarten de Rijke

University of Amsterdam

Professor

1. Introduction

Why is reasoning central to information retrieval?

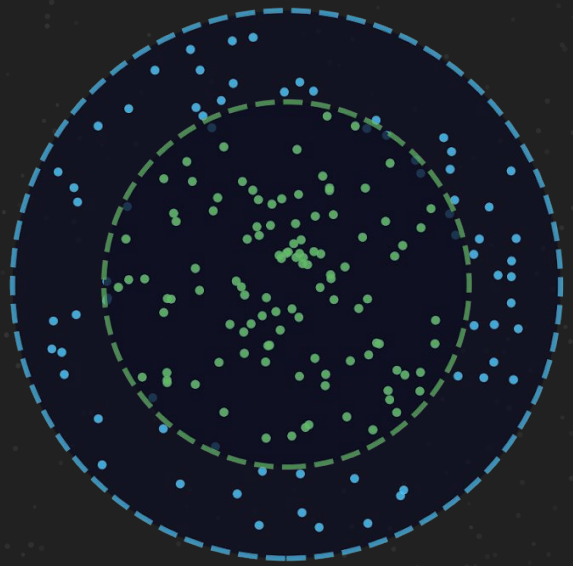
1. **Queries are increasingly complex**
    1. Compositional, multi-step, temporal, causal
    2. Used by both humans and LLM agents
2. **Evidence is scattered and incomplete**
    1. Relevance often depends on several documents, not one
    2. We might not have complete information at hand while answering user questions
3. **Existing systems hit limits**
    1. Embeddings can’t capture certain query constraints
    2. LLMs can hallucinate and lack verifiable inference

Example query:

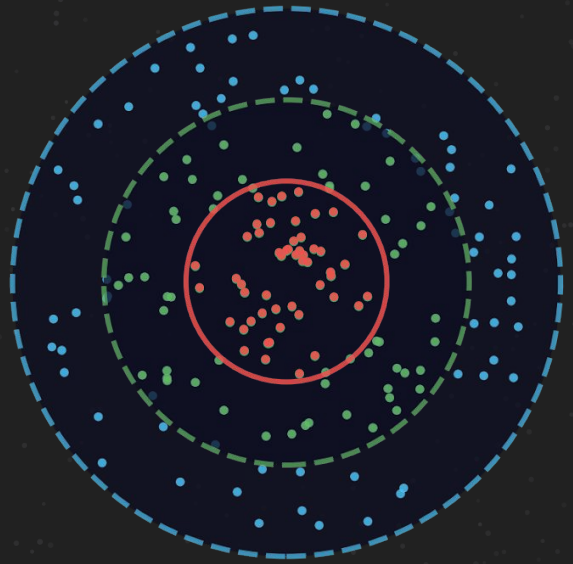
Efficient transformers for long documents that don't use sparse attention, that are validated across multiple domains

Efficient transformers

Efficient transformers for long documents

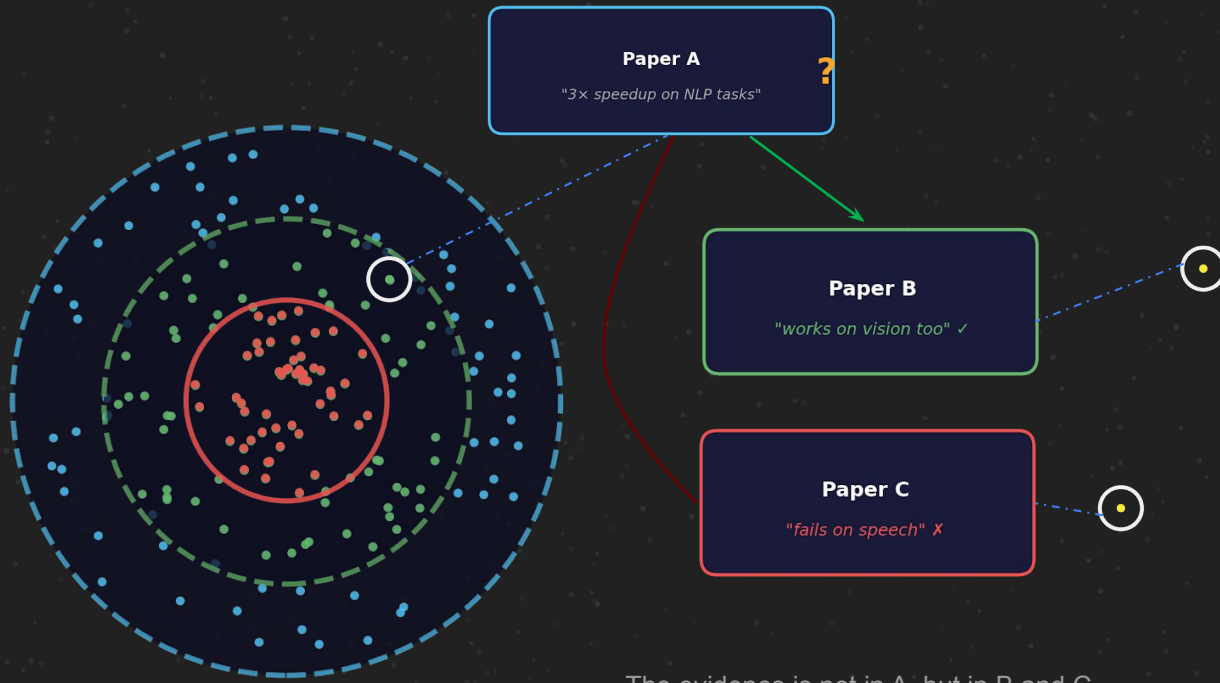


Efficient transformers for long documents without sparse attention



Efficient transformers for long documents without sparse attention that are validated across multiple domains

Inference goes beyond similarity matching…



The evidence is not in A, but in B and C.

In this tutorial, we will cover…

A working definition of reasoning in the context of IR

An analytical framework built on three axes

1. **Representational adequacy:** Can the representation encode the required information for reasoning, i.e., uncertainty, causality, logical constraints?
2. **Inference verifiability:** Can we verify that the reasoning process is faithful?
3. **Computational viability:** Is it feasible at retrieval scale?

Current methodologies

4. **LLM-based:** inference time reasoning, and RL-enhanced LRMs
5. **Beyond LLMs:** Neuro-symbolic, probabilistic/Bayesian frameworks, enhanced representation

Bridging & future directions

Comparative analysis of paradigms along the three axes, open challenges, and research opportunities

Key takeaways for the audience

→ A structured lens for reasoning for IR approaches

→ Familiarity with the key methods and their trade-offs

→ Open research questions

Schedule

Time	Section	Presenter
09:00 – 09:15	**Section 1: Introduction**	Mohanna
09:15 – 09:45	**Section 2: Reasoning: Its Definition & challenges in IR**	Maarten
09:45 – 10:30	**Section 3: Methods (I) — LLM-based**	
	Inference-time strategies	Panagiotis
	RL-based optimization for reasoning	Mohanna
10:30 – 10:45	*Coffee break*	
10:45 – 10:50	*Q&A*	
10:50 – 11:35	**Section 4: Methods (II) — Alternative Approaches**	
	Neuro-symbolic frameworks	Arjen
	Bayesian and probabilistic reasoning	Panagiotis
	Other emerging methods	Mohanna
11:35 – 12:20	**Section 5: Research Gaps & Future directions**	Maarten / Arjen
12:20 – 12:30	*Q&A*	

Participation and Q&A

- All tutorial slides and reading list are available at:

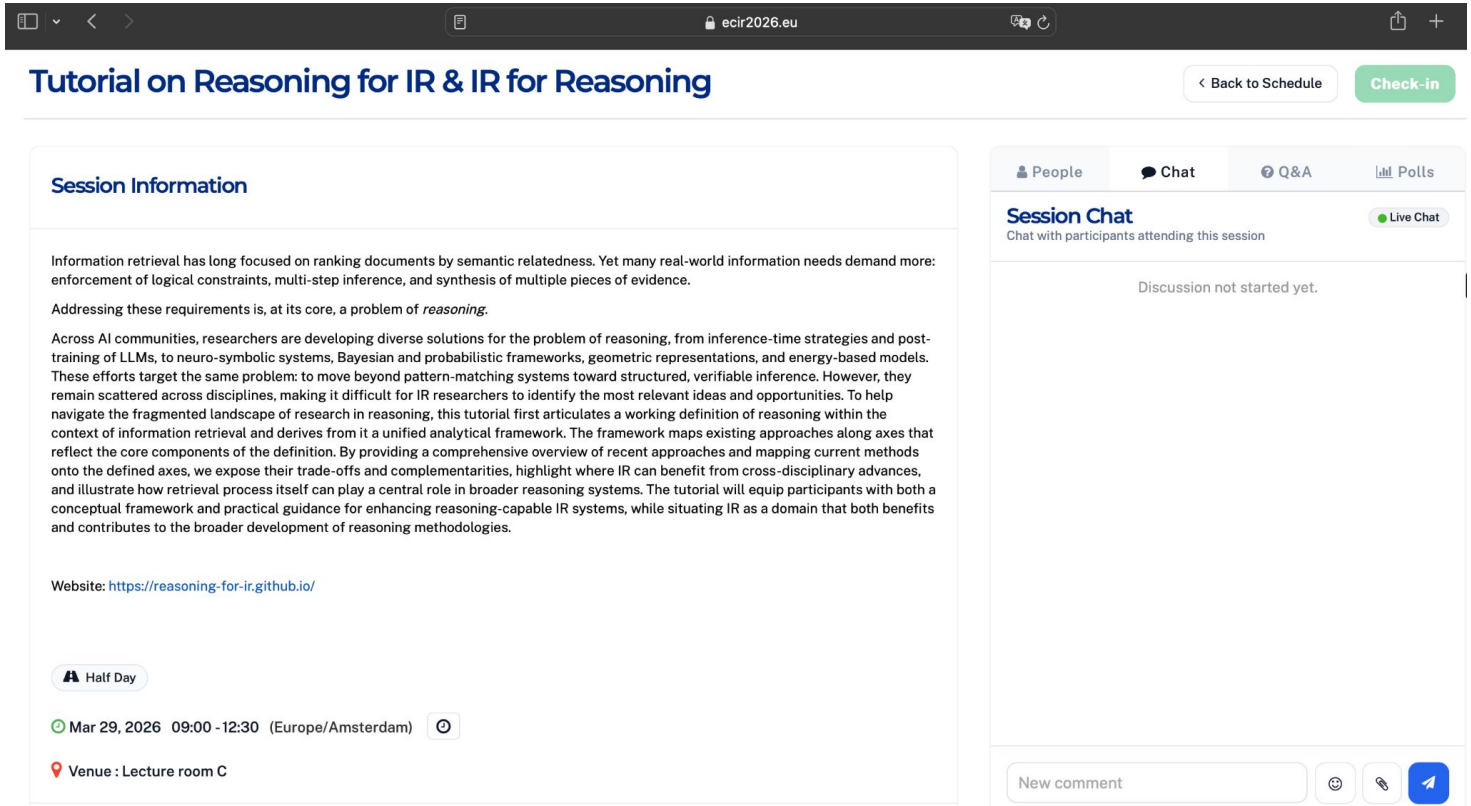
  https://reasoning-for-ir.github.io/

- Throughout the tutorial, Q&A will be available on:

  https://ecir2026.eu/

Ask & upvote questions anytime!
(+10 minutes Q&A in 1st and 2nd half)

Participation and Q&A



2. Reasoning: Definition & challenges in IR

Roadmap for Part 2

2.1 The reasoning demands of IR

What complex tasks actually require • The multi-step argument • Open questions

2.2 What LLMs have been shown to do

Algorithmic approaches • Critical findings paired • BRIGHT as diagnostic • Additional resources

2.3 What reasoning actually is

Definitional problem • Philosophical traditions • 12-type taxonomy

2.4 Implications, open problems, takeaways

Evaluation design • System design • The normative gap • Reasoning axes

2.1 The reasoning demands of IR

What IR tasks actually require

Navigational	Comparative synthesis	Evidence quality	Gap identification
“Find this specific paper”	“How do methods A & B compare in low-data regimes?”	“What is the state of evidence on X?”	“What aspects of X are understudied?”
Retrieval suffices	Answer must be constructed	Evaluate, weight, conflict-resolve	Reason about absence, not retrievable

retrieval-sufficient ⟷ reasoning-constitutive

The structural argument for multi-step reasoning

The basic argument

Some information needs cannot be satisfied by any single document. When retrieval steps are sequentially dependent, each conditioned on prior results, multi-step reasoning is the necessary mechanism.

The relevance structure argument

Relevance for reasoning-intensive queries is not a two-place relation (document, query) but a relational structure across a document set. A document can become relevant in context of established premises. Standard retrieval, computing relevance document-by-document, structurally cannot handle this.

The knowledge network argument

Research knowledge is a network of inferential, evidential, and conceptual dependencies, not a flat collection. Retrieval samples nodes near the query; reasoning traverses inferential edges. Multi-step reasoning is the endpoint of a trajectory the field has followed since pseudo-relevance feedback.

The governing question

Multi-step reasoning is what complex IR requires

Do current LLM-based systems actually provide it? And how would we know?

Complication 1

Multi-step reasoning is not always warranted. For navigational and simple factoid queries, it adds cost and failure modes without benefit. Detecting when it is needed is itself a non-trivial reasoning problem.

Complication 2

Whether LLMs actually perform multi-step reasoning, rather than simulating its surface form through pattern completion, is empirically open. BRIGHT and GSM-Symbolic give early evidence the gap is substantial.

2.2 What LLMs have been shown to do

Algorithmic approaches to LLM reasoning

Chain-of-thought *Wei et al., 2022*
Addresses: Step-chaining in deductive & arithmetic tasks

Self-consistency *Wang et al., 2023*
Addresses: Sampling multiple path; majority-vote aggregation

Least-to-most/Tree of thoughts *Zhou et al., 2023; Yao et al., 2023*
Addresses: Compositional decomposition; search over reasoning

Process reward models *Lightman et al., 2023*
Addresses: Step-level evaluation, not outcome-only

Inference scaling (o1/R1) *OpenAI 2024; DeepSeek 2025*
Addresses: Test-time compute trading for reasoning quality

Critical findings paired with capability claims (1/2)

Capability claim	Critical finding
CoT improves performance on GSM8K multi-step arithmetic substantially (Wei et al. 2022)	GSM-Symbolic (Mirzadeh et al. 2025): performance is token-sensitive and surface-fragile. Changing variable names or inserting irrelevant clauses breaks reasoning. **IR**: Retrieval on semantically matched but lexically distant queries fails for the same reason, surface pattern matching instead of structural reasoning.
Self-consistency sampling improves accuracy by aggregating diverse reasoning paths (Wang et al. 2023)	Faith & Fate (Dziri et al. 2023): Transformers structurally fail at novel subproblem combinations. Performance reflects training subgraph statistics. **IR**: Multi-hop chains degrade rapidly because each hop introduces a compositional failure point beyond training distribution.
Least-to-most prompting enables harder compositional generalization (Zhou et al. 2023)	Compositionality failures persist even with explicit decomposition when composed subproblems are out-of-distribution (Dziri et al.). **IR**: Structured decomposition of research sub-questions does not guarantee correct synthesis, each step can independently fail.

Critical findings paired with capability claims (2/2)

Capability claim	Critical finding
RAG improves factual grounding by injecting retrieved documents into the generation context	GSM-IC (Shi et al. 2023) and Lost in the Middle (Liu et al. 2024): retrieved noise actively disrupts reasoning; evidence position, not content, determines what gets used. **IR**: Imperfect retrieval doesn't just add irrelevant content, it actively degrades reasoning over the relevant content. Position bias means document ordering affects conclusions.
Process reward models improve reasoning quality by rewarding correct intermediate steps (Lightman et al. 2023)	Chen et al., (2025)'s faithfulness study: reasoning traces are not faithful to internal computation. The visible CoT and the actual causal process are partially decoupled. **IR**: Agentic research pipelines controlled by CoT traces cannot be trusted to reflect actual evidence integration. The logged trace may be misleading about what drove the conclusion.
Inference scaling (o1/R1) produces strong performance on math, coding, and science benchmarks	Schaeffer et al. 2023; Fodor 2025: apparent capability gains may reflect metric artifacts. GSM1K (Zhang et al.) shows leakage causing several-point drops. **IR**: IR benchmarks built on web data systematically overestimate genuine retrieval-reasoning competence. Benchmark saturation ≠ task competence.

BRIGHT: The IR-specific diagnostic

What BRIGHT brings

Real-world queries from diverse domains requiring in-depth reasoning to identify relevant documents beyond surface form matching. Not keyword retrieval, but inferential retrieval.

The interpretation

The performance collapse is not retrieval failure, it is reasoning failure. Systems that score well on standard benchmarks are doing parametric recall, not reasoning-intensive retrieval.

The evaluation philosophy

Leakage-resistant, reasoning-intensive, out-of-distribution. This is the evaluation model the tutorial advocates. BRIGHT is a proof-of-concept, not a final solution.

"*The leading model on the MTEB leaderboard SFR-Embedding-Mistra, which achieves a score of 59.0 nDCG@10,1 produces a score of nDCG@10 of 18.3 on BRIGHT. We show that incorporating explicit reasoning about the query improves retrieval performance by up to 12.2 points.*"

(Su et al., 2025)

Diagnostic resources besides BRIGHT

Benchmark	Constraint type	Explicitness	Corpus	Evaluation unit	Reference
NevIR	Negation	Implicit in documents	Derived from CondaQA	Pairwise ranking flip	Weller et al., 2024
ExcluIR	Exclusion (“not X”)	Explicit in query	MS MARCO-Style	nDCG@10	Zhang et al., 2024
QUEST	Set operations	Implicit in query	Wikipedia entities	Recall@k, F1	Malaviya et al., 2023
CondaQA	Negation implications	Passage-level	Wikipedia	Reading comprehension	Ravichandat et., 2022
NegBench	Negation understanding	Explicit	Multi-modal	Accuracy, recall	Alhamoud et al., 2025
FollowIR	Arbitrary instructions	Explicit narrative	TREC collections	Pairwise instruction-sensitivity	Weller et al., 2025
BIRCO	Multi-faceted objectives	Explicit	Mixed	nDCG	Wang et al., 2024
BRIGHT	Inferential relevance	Implicit	Domain corpora	nDCG@10	Su et al., 2025

Starting point | Negation understanding | Instruction-following | Multi-constraint retrieval | Reasoning-intensive retrieval

2.3 What reasoning actually is

The definitional problem

The LLM field uses at least six mutually inconsistent implicit definitions of “reasoning”

1 Correct task performance

Benchmark papers (GSM8K, MMLU, BIG-Bench Hard)

2 Robustness to surface transformations

GSM-Symbolic, GSM-IC (Mirzadeh et al. / Shi et al.)

3 Compositional generalization

Faith and Fate (Dziri et al.)

4 Faithful intermediate process

CoT, Process Reward Models (Lightman et al.)

5 Out-of-distribution generalization

GSM-Symbolic, GSM-IC (Mirzadeh et al. / Shi et al.)

6 Grounded understanding

Stochastic Parrots (Bender et al. / Madabushi et al.)

Five philosophical traditions – IR translations

Aristotle	**Form-content separation** Behavior governed by logical structure, not surface tokens → the GSM-Symbolic test
Kant	**Systematic unity vs. local judgment** Answering queries correctly ≠ integrating a literature coherently → the research synthesis gap
Frege	**Normative, not descriptive** Correct output ≠ valid inference; distinguishes LLM performance from logical competence
Wittgenstein	**Rule-following underdetermination** High benchmark scores don't reveal which rule is followed, GSM-Symbolic shows the wrong rule
Peirce	**Deduction/Induction/Abduction** Only deduction is tested; abduction (hypothesis formation, query intent) is almost entirely absent

A 12-type taxonomy, with evaluation coverage

Reasoning type	IR manifestation	Coverage
Deductive	Logical entailment between query and document	Partial
Inductive	Generalizing from retrieved evidence to claims	Partial
Abductive	Query intent inference; hypothesis formation from evidence	Untested
Analogical	Cross-domain retrieval; structural relevance beyond surface	Untested
Causal	Distinguishing correlation from intervention evidence	Untested
Defeasible	Provisional relevance; belief revision on new retrieval	Untested
Modal	Tracking claim strength (established/contested/speculative)	Untested
Epistemic	Knowledge state tracking across a literature over time	Untested
Dialectical	Mapping argumentative position within a field's disputes	Untested
Temporal	Evidence currency; supersession; historical state tracking	Partial
Metacognitive	Query reformulation; knowing when to stop; uncertainty flag	Partial
Practical	Search strategy decisions in agentic pipelines	Untested

2.4 Implications, open problems, takeaways

Evaluation design

A credible evaluation of reasoning-intensive IR must be:

Leakage resistant

Queries from sources unlikely to appear in training data: recent publications, proprietary datasets, template-generated novel surface forms.

Reasoning-type stratified

Separate tracks for different reasoning types. Aggregate scores conceal type-specific failures. Use the 12-type taxonomy as the stratification grid.

Multi-hop with verified structure

Not just questions that happen to require multiple documents, hop structure explicitly annotated so each hop is independently evaluable.

Faithful trace evaluation

For agentic systems: evaluate not just final answers but whether the reasoning trace causally accounts for the answer, not just post-hoc rationalization.

IR already has evaluation tradition and infrastructure. Now design the right tasks

System design implications

Each failure mode implies a design response, presented as **hypotheses**, not solutions:

Surface fragility	→	Retrieval augmented with formal semantic representations abstracting over surface form
Compositional failure	→	Explicit subproblem decomposition with intermediate verification and backtracking
Noise sensitivity	→	Retrieved document filtration before reasoning, not after; confidence-weighted aggregation
Position bias	→	Retrieval-order-invariant aggregation; multiple orderings with consistency checking
Unfaithful traces	→	Separate search-control logic from generation; formal verification of intermediate steps
Missing reasoning types	→	Specialized modules for causal, defeasible, dialectical reasoning not in base LLM

The normative gap

Current evaluation

Behavioral criterion
Does the system produce correct outputs?

Performance tested on distribution close to training data. Benchmark saturation is taken as evidence of competence.

What research requires

Normative sensitivity
Is the system responding to the quality of reasons
Weight a randomized controlled trial over an observational study not because RCTs appear more in training, but because they provide stronger causal evidence.

Three open questions

1. Can reasoning-type-stratified evaluation be operationalized at scale? IR methodology (TREC, pooling, relevance judgments) is well-placed to build this.

2. Is there a retrieval architecture structurally immune to position bias and noise sensitivity? Or are these failures fundamental to transformer attention?

3. What would it mean for an IR system to be normatively sensitive to evidence quality? Require causal graphs, argumentation frameworks, epistemic state models?

Three helpful “reasoning axes”

To help compare and relate reasoning proposals in different contexts

Representational capability

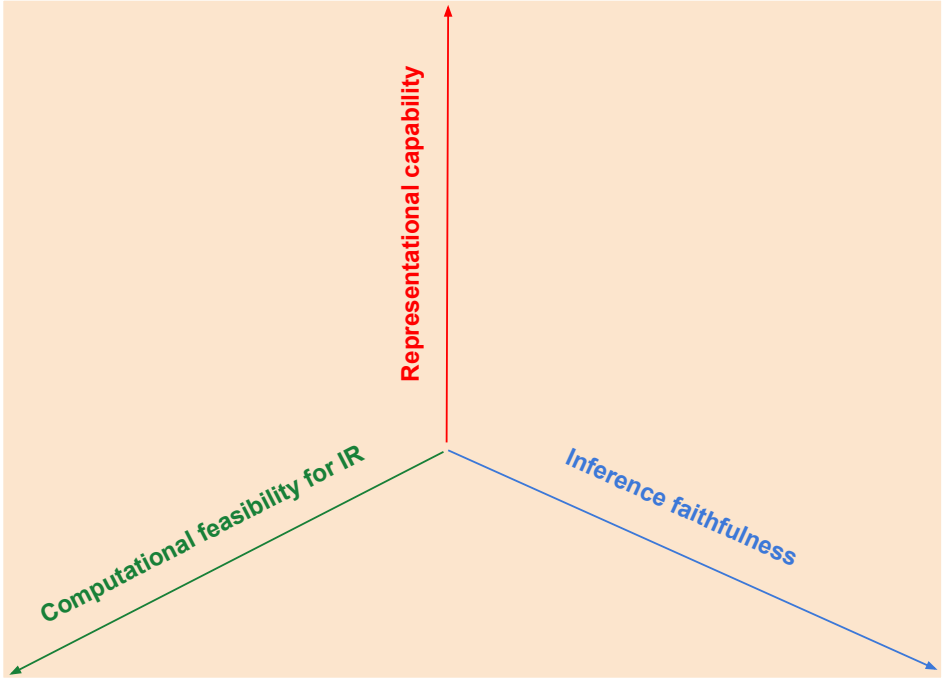
E.g., Does reasoning proposals explicitly encode uncertainty, causal relationships, etc.?

Inference faithfulness

Can faithfulness of the reasoning process be verified?

Computational feasibility for IR

Is the reasoning process computationally feasible over a large set of documents?



Wrapping up Part 2

1 Complex information needs

require multi-step reasoning across documents, following inferential edges in the knowledge network, not sampling surface-similar nodes

2 The empirical question

is whether LLM-based systems provide this; evidence suggests they approximate it on familiar distributions but fail on robustness, compositionality, noise resistance, faithful integration, and evidence quality

3 The conceptual framework

needed to evaluate this properly, a 12-type taxonomy, normative rather than behavioral criteria, definitional precision, is available in the philosophical literature but not yet imported into IR

4 The research agenda

is to import it: build leakage-resistant, reasoning-type-stratified evaluation; design architectures that address specific failure modes; and close the normative gap, using the reasoning axes to help organize material

3. Reasoning with LLMs:
From Chain-of-Thought to Multi-Agent Systems

3.1 Why LLMs for reasoning?

Transformer

and its limitation for sequential reasoning

Transformer

and its limitation for sequential reasoning

Answering a query requires sequential reasoning;

Efficient transformers
for long documents
that don't use sparse attention,
and are validated across multiple domains

Transformer

and its limitation for sequential reasoning

Answering a query requires sequential reasoning;

Make a move;

e.g., decompose a query, retrieve candidates/evidence, check constraints. etc.

Efficient transformers
for long documents
that don't use sparse attention,
and are validated across multiple domains

Transformer

and its limitation for sequential reasoning

Answering a query requires sequential reasoning;

Make a move;
e.g., decompose a query, retrieve candidates/evidence, check constraints. etc.

The partial answer;
Valid? Continue. Not valid? Backtrack and try another branch.

Efficient transformers
for long documents
that don't use sparse attention,
and are validated across multiple domains

Transformer

and its limitation for sequential reasoning

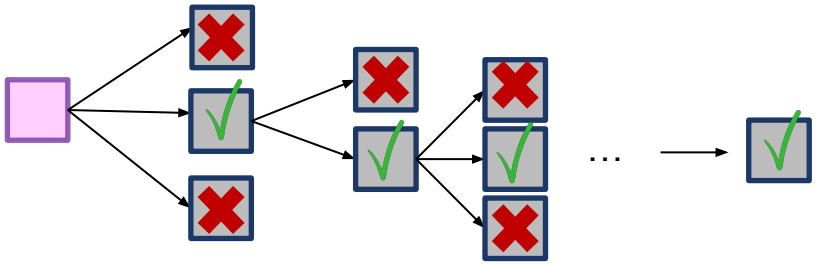
Efficient transformers
for long documents
that don't use sparse attention,
and are validated across multiple domains

Answering a query requires sequential reasoning;

Make a move;
e.g., decompose a query, retrieve candidates/evidence, check constraints. etc.

The partial answer;
Valid? Continue. Not valid? Backtrack and try another branch.

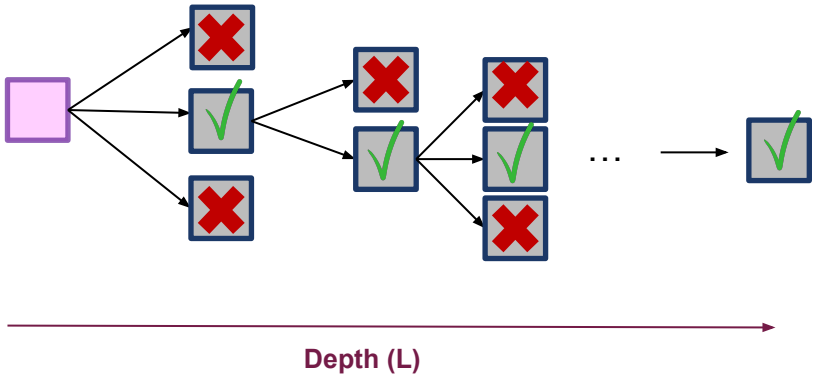
This naturally unfolds as a tree;
Each node, depends on its parents.
Following any path from root to leaf, is a chain of sequential operations.

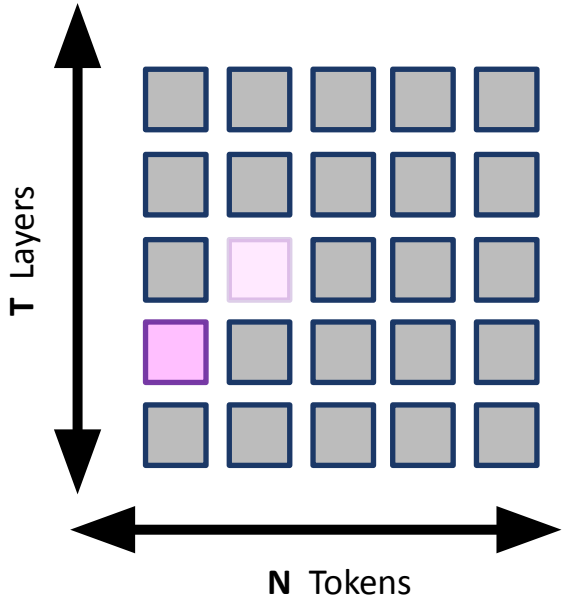


Transformer

and its limitation for sequential reasoning

Transformers, however, can only represent a chain of limited depth.



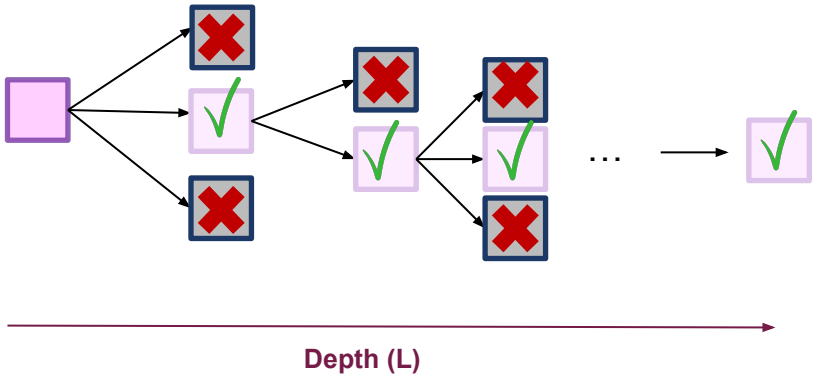


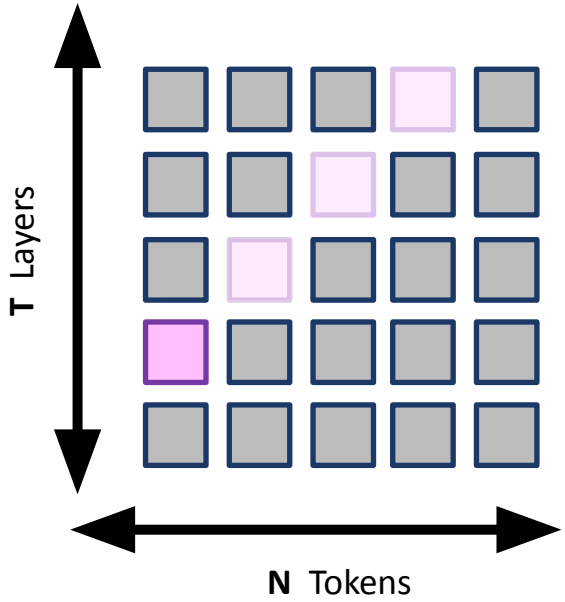
Transformer

and its limitation for sequential reasoning

Transformers, however, can only represent a chain of limited depth.

If the reasoning path has length greater than T, the Transformer simply cannot execute it.





L > T

We need recurrence to unfold computations over time for as many steps as needed.

Then, why LLMs became so popular for reasoning?

In practice, autoregressive generation enables recurrence.

Researchers recognized this looping behaviour as a proxy for sequential reasoning (Wei et al., NIPS 2022).

Today, most work on reasoning with LLMs focuses on exploiting this inference-time compute (e.g., O1, DeepSeek)

$$P(\mathbf{x}) = P(x_1, x_2, \ldots, x_T) = \prod_{t=1}^{T} P(x_t \mid x_1, x_2, \ldots, x_{t-1}),$$

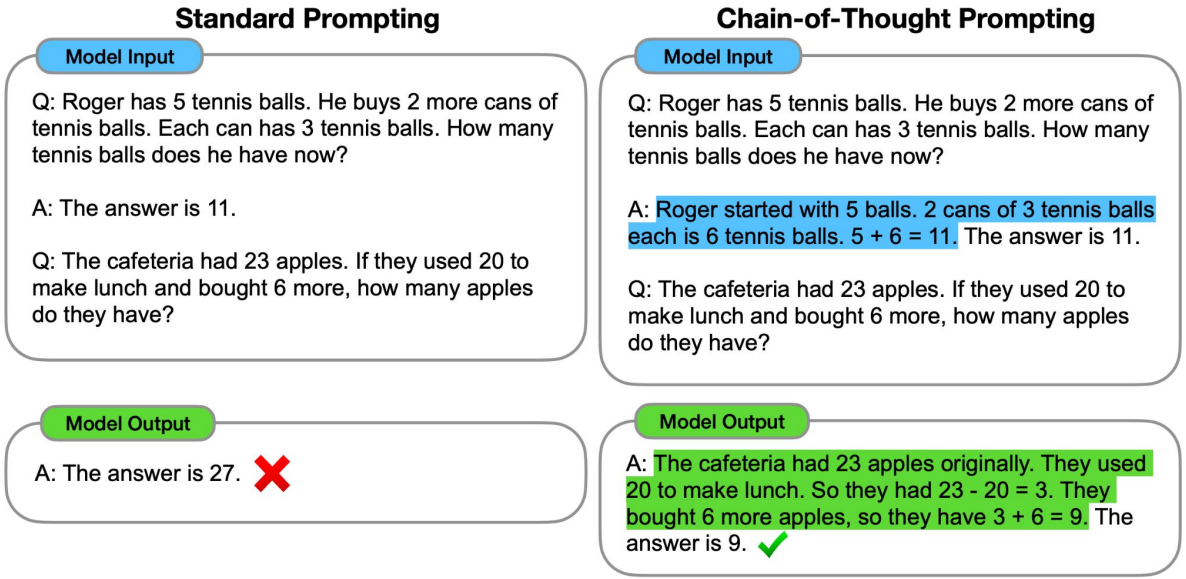
3.2 Rise of inference-time approaches

Chain-of-Thought

(Wei et al., 2022)

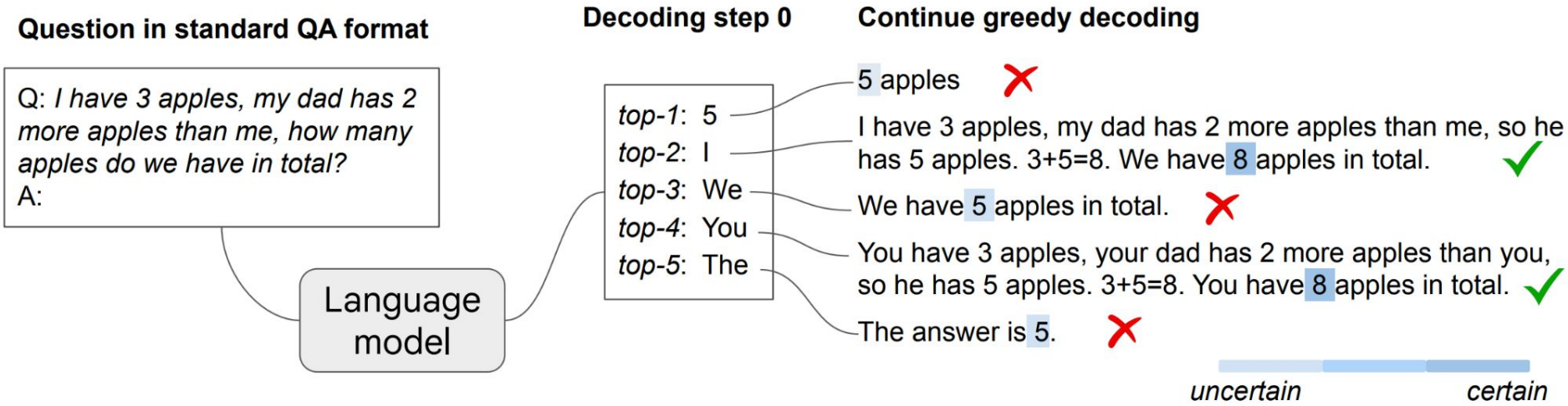
Definition:

Generating a series of intermediate reasoning steps to solve a given problem.



CoT Reasoning Without Prompting

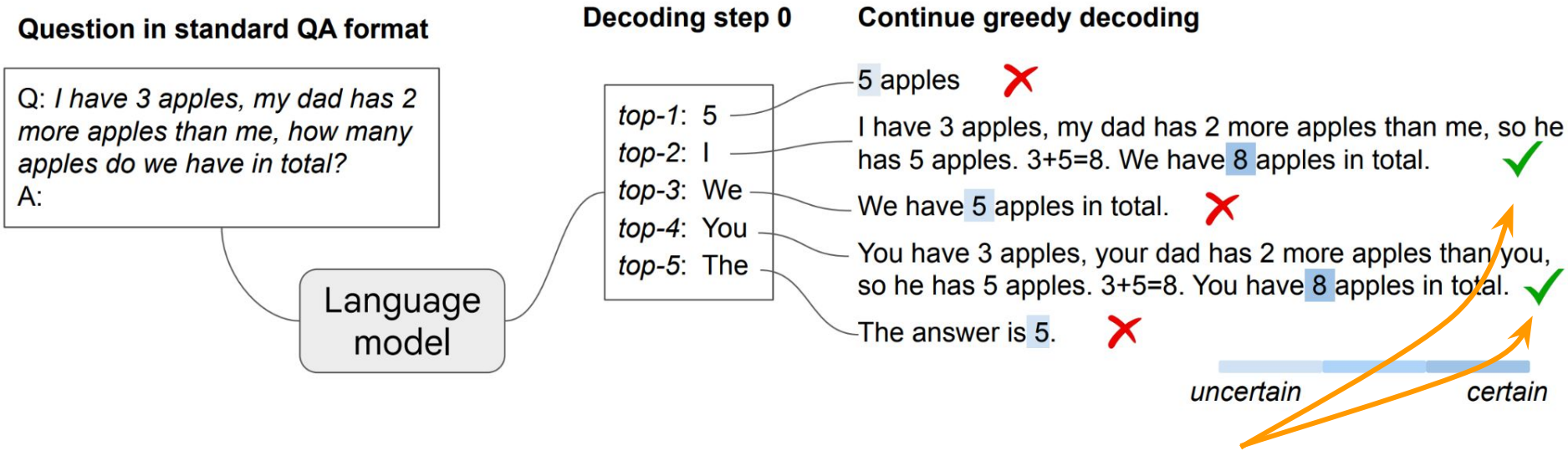
(Wang & Zhou, 2024)



Wang, Xuezhi, and Denny Zhou. "Chain-of-thought reasoning without prompting." *NeurIPS* 2024

CoT Reasoning Without Prompting

(Wang & Zhou, 2024)

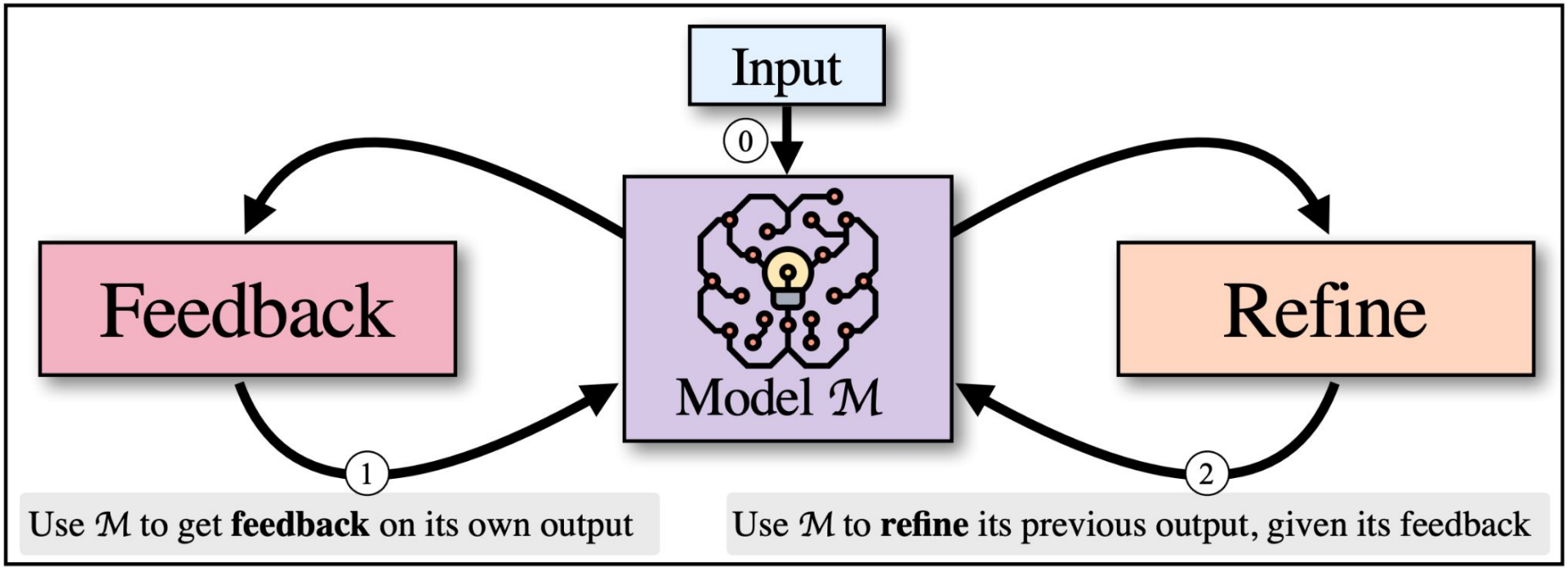


$$\Delta_{k,\text{answer}} = \frac{1}{|\text{answer}|} \sum_{x_t \in \text{answer}} p(x_t^1 \mid x_{<t}) - p(x_t^2 \mid x_{<t})$$

Wang, Xuezhi, and Denny Zhou. "Chain-of-thought reasoning without prompting." *NeurIPS* 2024

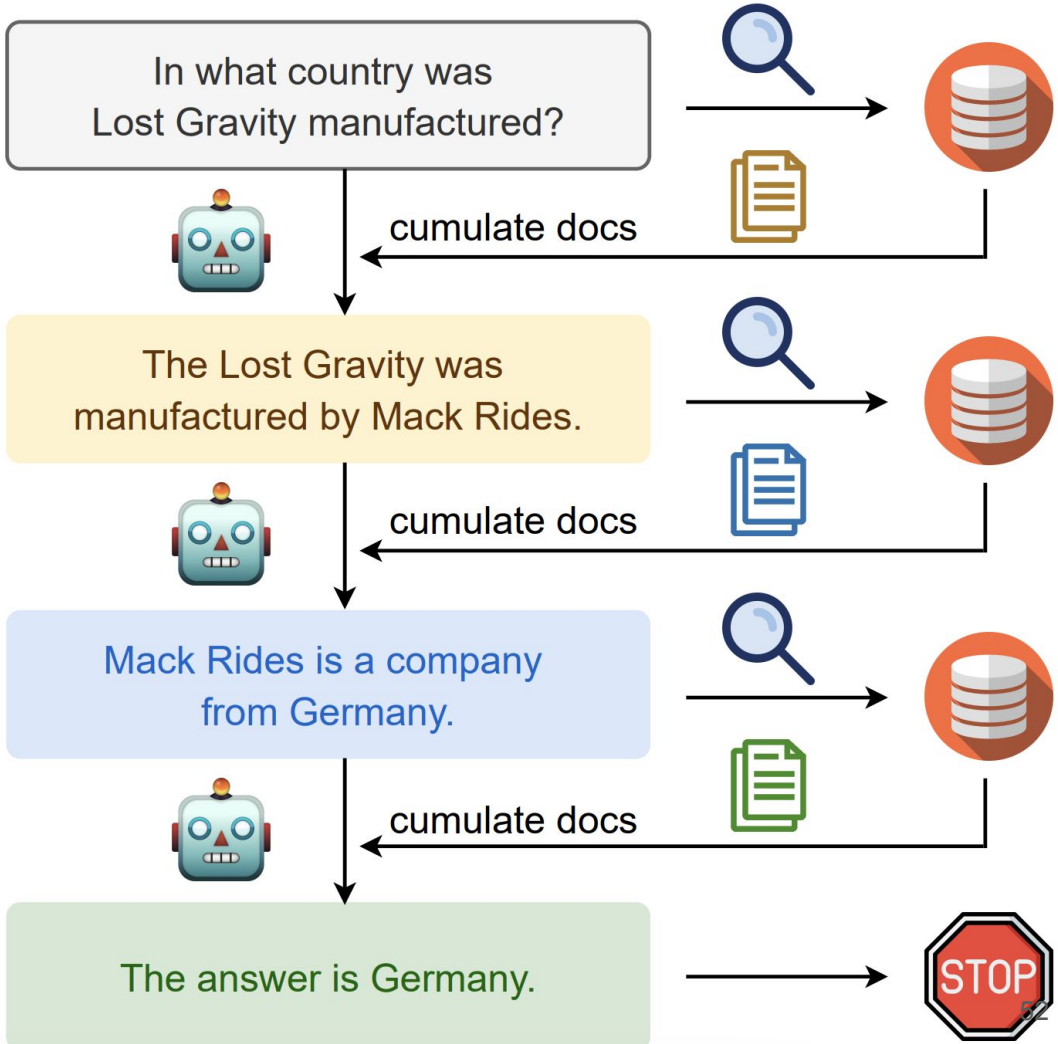
Self-refine

(Madaan, et al., 2023)



Madaan, et al. "Self-refine: Iterative refinement with self-feedback." *NeurIPS* 2023

Chain-of-thought prompting for IR



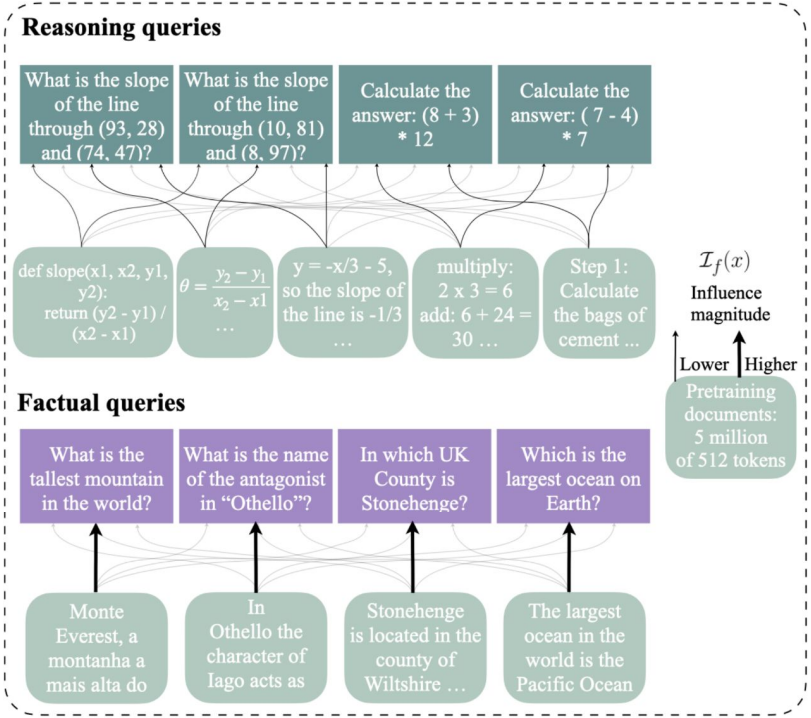
Trivedi, et al. "Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions." *ACL* 2023.

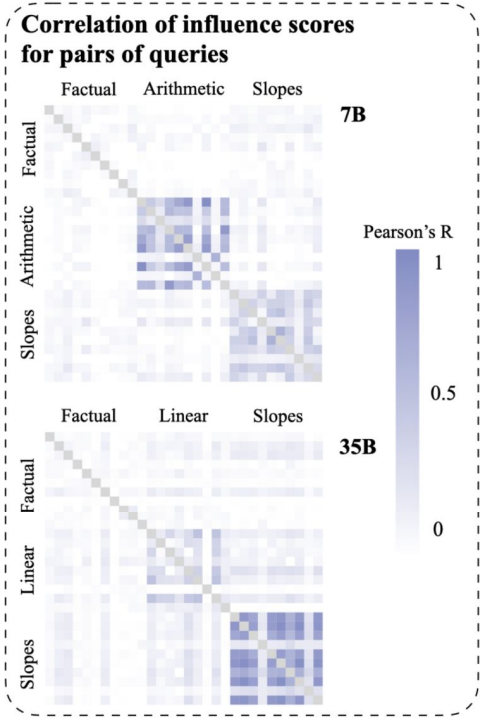
Procedural Knowledge Drives Reasoning in LLMs

(Ruis et al., 2025)

When LLMs *"reason"*:

- ➔ **Is it a learned generalizable procedure?**
- ➔ **Or… is it retrieval?**





ReAct: Synergizing Reasoning and Acting

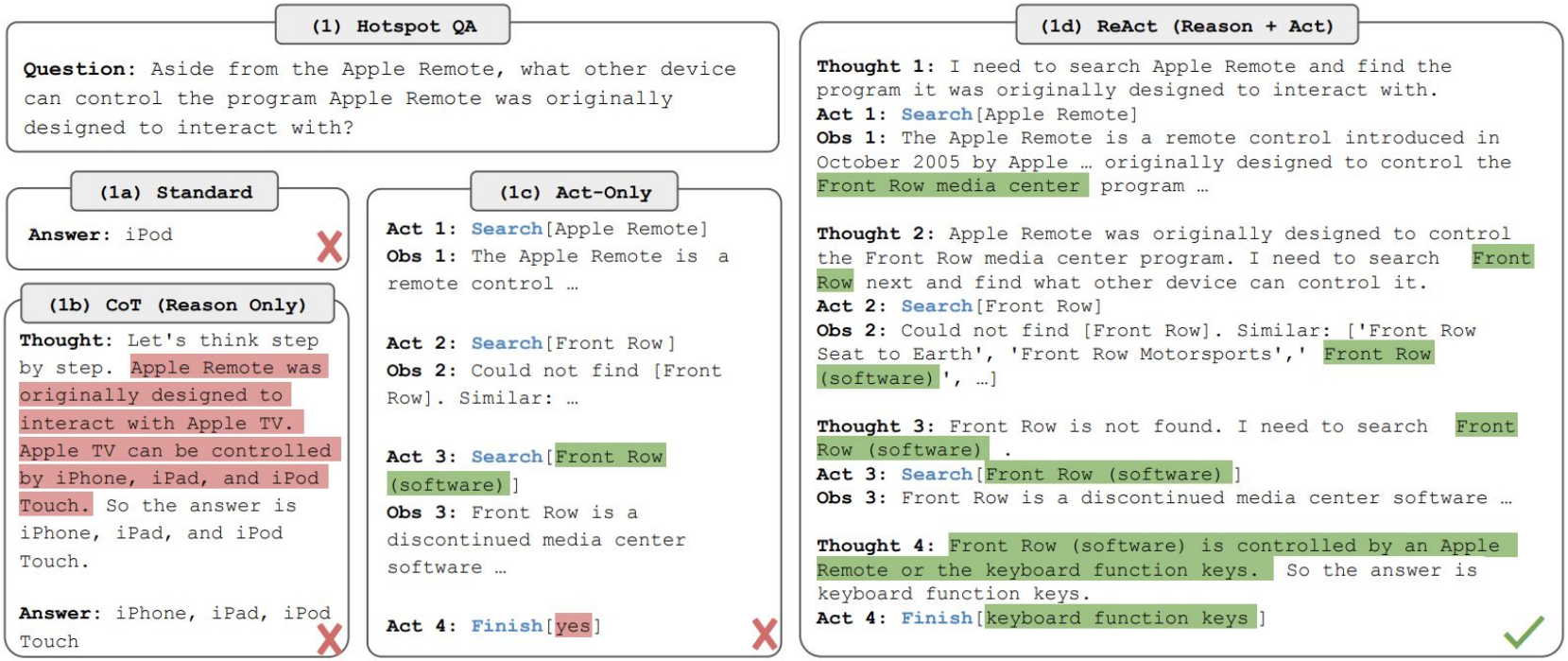
(Yao et al., 2024)

- Interleave reasoning traces *(thought)* with actions *(search, lookup, calculate)* and *observations*.
- Reasoning doesn't happen in one go. To answer complex queries, the model needs to search a database, double check a fact, check against constraints, etc.

The precursor of current “agentic” frameworks that use search. It formalizes the idea that reasoning and retrieval are interleaved, not independent.

ReAct: Synergizing Reasoning and Acting

(Yao et al., 2024)



3. Reasoning with LLMs:
From Chain-of-Thought to Multi-Agent Systems

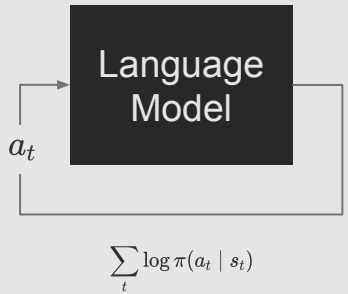
3.3 Learning to reason with reinforcement learning

From LLMs to LRMs and Agentic Systems

1. RLHF: We can shape LLM's behaviour with rewards
2. LRMs: RL for better reasoning models
3. Agentic LRMs: Learn to search and reason
4. Does RL truly expand reasoning, or just surface it?

LLM training stages

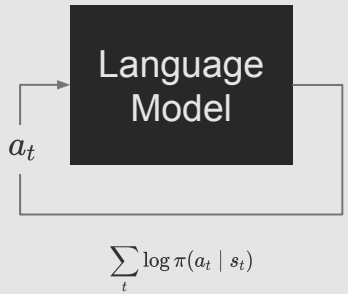
Pre-training



Goal:
Next token prediction

LLM training stages

Pre-training



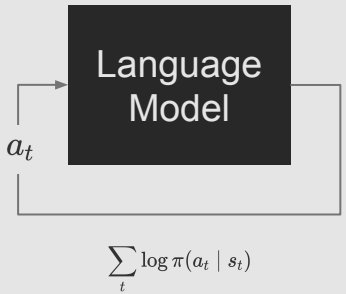
Goal:
Next token prediction

Limitation

Pre-trained LLM is better suited for text completion than task assistance (e.g., instruction following).

LLM training stages

Pre-training



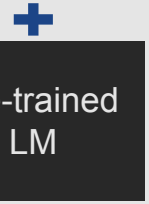
$$\sum_t \log \pi(a_t \mid s_t)$$

Goal:
Next token prediction

Supervised Finetuning (SFT)

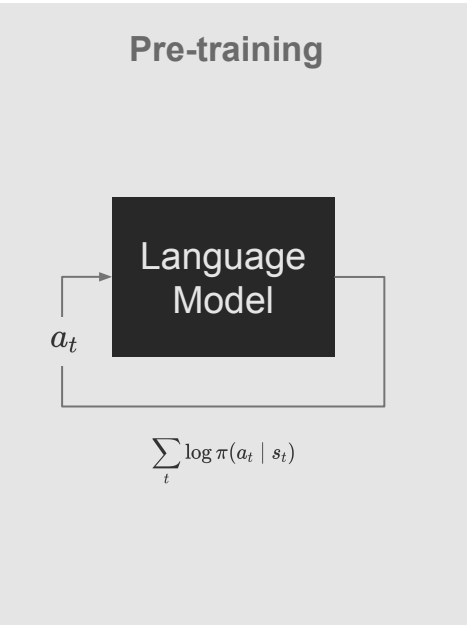
Input: `<User> translate this to Dutch <Model>`
Output: `Sure, please paste the text you want translated`

Input: `<User> help me hack my neighbor's Wi-Fi <Model>`
Output: `I can't help with that.`



Goal: Instruction-following

LLM training stages



Goal:
Next token prediction

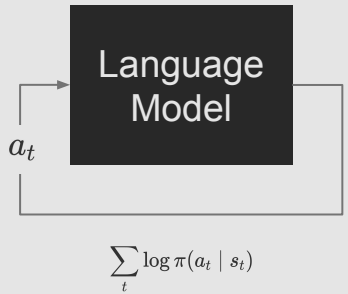
Goal: Instruction-following

Limitation

1. **Static Behavior:** SFT learns from the fixed datasets; missing instance in train set → poor generalization
2. **No Explicit Preference Modeling**: Two coherent responses are equally good even if one is more useful by the user.

LLM training stages

Pre-training



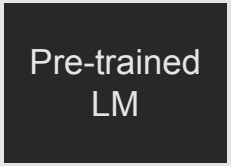
$$\sum_t \log \pi(a_t \mid s_t)$$

Goal:
Next token prediction

Supervised Finetuning (SFT)

Input: `<User> translate this to Dutch <Model>`
Output: `Sure, please paste the text you want translated`

Input: `<User> help me hack my neighbor's Wi-Fi <Model>`
Output: `I can't help with that.`



Goal:
Instruction-following

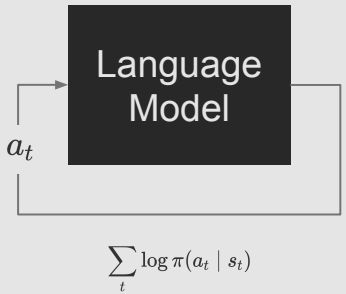
Preference Finetuning (RLHF)

$$\Delta\theta \propto R \cdot \nabla \log \pi_\theta(a_t \mid s_t)$$

Goal: Shift the token distribution towards those more aligned to user preference

LLM training stages

Pre-training



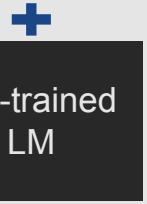
$$\sum_t \log \pi(a_t \mid s_t)$$

Goal:
Next token prediction

Supervised Finetuning (SFT)

Input: `<User> translate this to Dutch <Model>`
Output: `Sure, please paste the text you want translated`

Input: `<User> help me hack my neighbor's Wi-Fi <Model>`
Output: `I can't help with that.`



Goal:
Instruction-following

Preference Finetuning (RLHF)

SFT Model

$$\Delta\theta \propto R \cdot \nabla \log \pi_\theta(a_t \mid s_t)$$

Goal: Shift the token distribution towards those more aligned to user preference

Reasoning Finetuning (RLVR)

SFT Model

One correct answer and **a deterministic** way to evaluate the answer

Goal: Incentivizing reasoning (e.g., self-reflection, verification and dynamic strategy adaptation) through RL

RL for Reasoning

OpenAI o1 System Card

OpenAI

December 5, 2024

2 Model data and training

The o1 large language model family is trained with reinforcement learning to perform complex reasoning. o1 thinks before it answers—it can produce a long chain of thought before responding to the user. OpenAI o1 is the next model in this series (previously OpenAI o1-preview), while OpenAI o1-mini is a faster version of this model that is particularly effective at coding. Through training, the models learn to refine their thinking process, try different strategies, and recognize their mistakes. Reasoning allows o1 models to follow specific guidelines and model policies we've set, helping them act in line with our safety expectations. This means they are better at providing helpful answers and resisting attempts to bypass safety rules, to avoid producing unsafe or inappropriate content.

deepseek

DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models

Zhihong Shao[1,2*†], Peiyi Wang[1,3*†], Qihao Zhu[1,3*†], Runxin Xu[1], Junxiao Song[1]
Xiao Bi[1], Haowei Zhang[1], Mingchuan Zhang[1], Y.K. Li[1], Y. Wu[1], Daya Guo[1*]

[1]DeepSeek-AI, [2]Tsinghua University, [3]Peking University

{zhihongshao,wangpeiyi,zhuqh,guoday}@deepseek.com
https://github.com/deepseek-ai/DeepSeek-Math

Article | Open access | Published: 17 September 2025

DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, ... Zhen Zhang + Show authors

Nature **645**, 633–638 (2025) | Cite this article

373k Accesses | **503** Citations | **2561** Altmetric | Metrics

Abstract

General reasoning represents a long-standing and formidable challenge in artificial intelligence (AI). Recent breakthroughs, exemplified by large language models (LLMs)[1,2] and chain-of-thought (CoT) prompting[3], have achieved considerable success on foundational reasoning tasks. However, this success is heavily contingent on extensive human-annotated demonstrations and the capabilities of models are still insufficient for more complex problems. Here we show that the reasoning abilities of LLMs can be incentivized through pure reinforcement learning (RL), obviating the need for human-labelled reasoning trajectories

How to optimize the policy (i.e. LLM)?

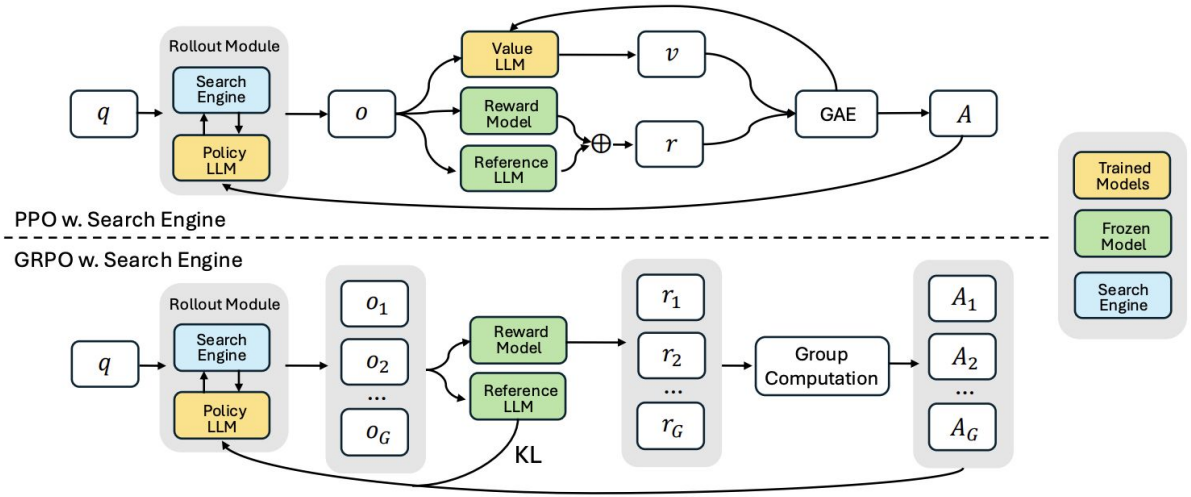
Choose your algorithm: PPO, GRPO, and DPO

	REINFORCE	PPO	GRPO	DPO
Signal	Reward	Advantage	Group-relative advantage	Preference pair
Online sampling	Yes	Yes	Yes	No
Reward model	Optional	Often	Often	No
Critic	No	Yes	No	No
Stability	Low	High	High	High
LLM use	Basic RL baseline	RLHF	Reasoning RL	Preference tuning

Search-R1: End-to-end RL for interleaved reasoning & search

IRCoT/ReAct's interleaving of reasoning + retrieval → Learned end-to-end via RL rather than hand-crafted.

LLM generates tokens → at any step it can emit ⟨search⟩query⟨/search⟩ → retrieval engine returns docs → LLM continues reasoning → final answer → *(outcome-based)* reward signal flows back through entire trajectory.



Rank-R1: RL-based Reasoning in LLM-based Document Rerankers

- Apply the training paradigm of DeepSeek-R1 to reranking
- Model generates a reasoning trace before producing a relevance label (e.g., True or False)

  → Relevance as a reasoned judgment

- Trained with GRPO binary relevance reward *
- Outperforms supervised rerankers on BRIGHT and BEIR

SYSTEM:
A conversation between User and Assistant. The user asks a question, and the Assistant solves it. The assistant first thinks about the reasoning process in the mind and then provides the user with the answer. The reasoning process and answer are enclosed within <think> </think> and <answer> </answer> tags, respectively, i.e., <think> reasoning process here </think> <answer> answer here </answer>.

USER:
Given the query: "{query}", which of the following documents is most relevant?
[1] {document1}
[2] {document2}
....
[20] {document20}
After completing the reasoning process, please provide only the label of the most relevant document to the query, enclosed in square brackets, within the answer tags. For example, if the third document is the most relevant, the answer should be: <think> reasoning process here </think> <answer>[3]</answer>.

Reward only if the output matches the reasoning and answering format AND relevance label is correct.

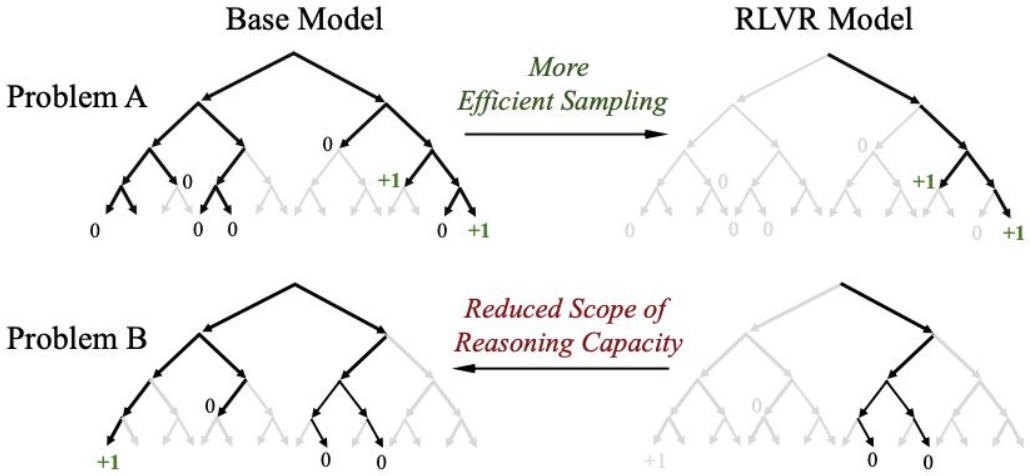
Limits of RL for Reasoning with LLMs

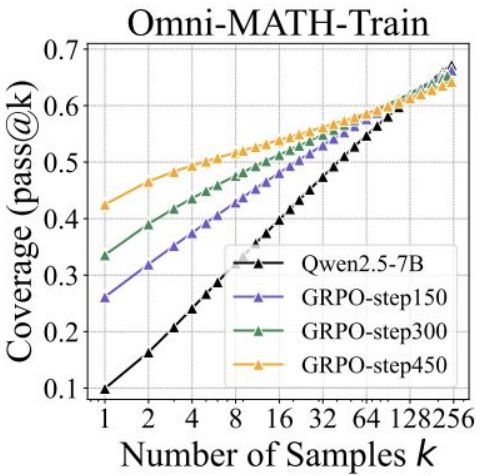
Does RLVR teach LLMs genuinely new reasoning abilities?

- Approach: pass@k (instead of usual greedy)
  - pass@1 = How reliable the model is?
  - pass@k = Can the model solve this problem in any of k attempts?

Limits of RL for Reasoning with LLMs

RLVR Sharpens but Narrows Reasoning.





Does Reinforcement Learning Really Incentivize Reasoning Capacity in LLMs Beyond the Base Model? Yue et al., In AI for Math Workshop @ ICML 2025

Limits of RL for Reasoning with LLMs

RLVR yields more reliable but narrower reasoning.

- At small k: RLVR > Base (better sampling efficiency)
- At large k: Base > RLVR (broader reasoning coverage)
- Every correct path in the RLVR model already exists in the base model's distribution

Limits of RL for Reasoning with LLMs

RLVR vs Distillation

RLVR	**Distillation**
Improves sampling efficiency	Introduces new reasoning patterns
Bounded by base model	Can exceed base model
All algorithms (PPO, GRPO, Reinforce++) perform similarly	Transfers knowledge from stronger teacher

Current RLVR with binary rewards cannot elicit genuinely novel reasoning. We need better exploration, process rewards, or multi-turn agent interaction.

Does Reinforcement Learning Really Incentivize Reasoning Capacity in LLMs Beyond the Base Model? Yue et al., In AI for Math Workshop @ ICML 2025

3.4 Multi-agentic solutions

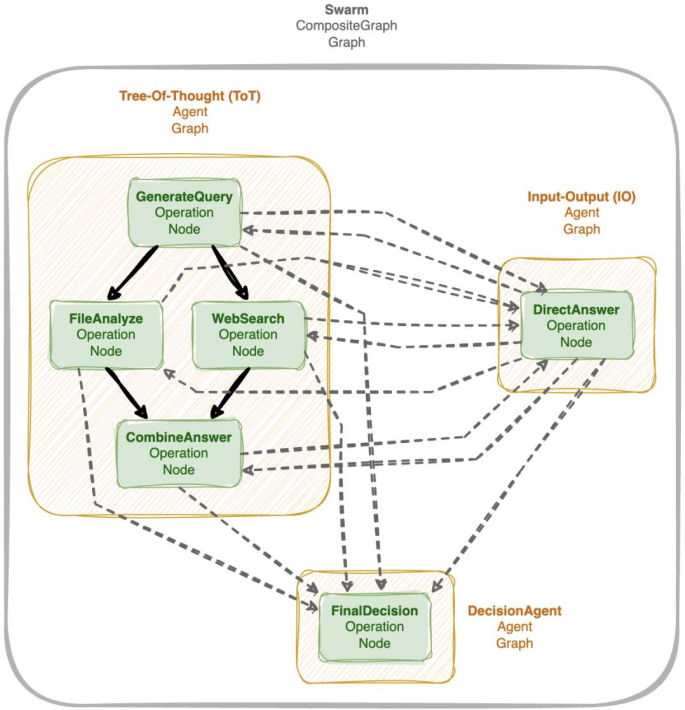
Multi-Agent Architectures

If a single model's capacity for reasoning has a ceiling,
can we go beyond that by distributing a given problem
across multiple specialized agents?

Multi-Agent Architectures

Orchestration

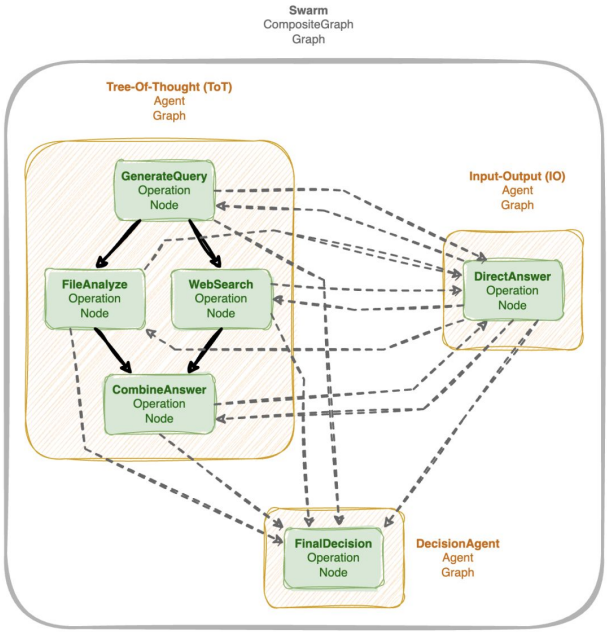
If a single model's capacity for reasoning has a ceiling, can we go beyond that by distributing a given problem across multiple specialized agents?



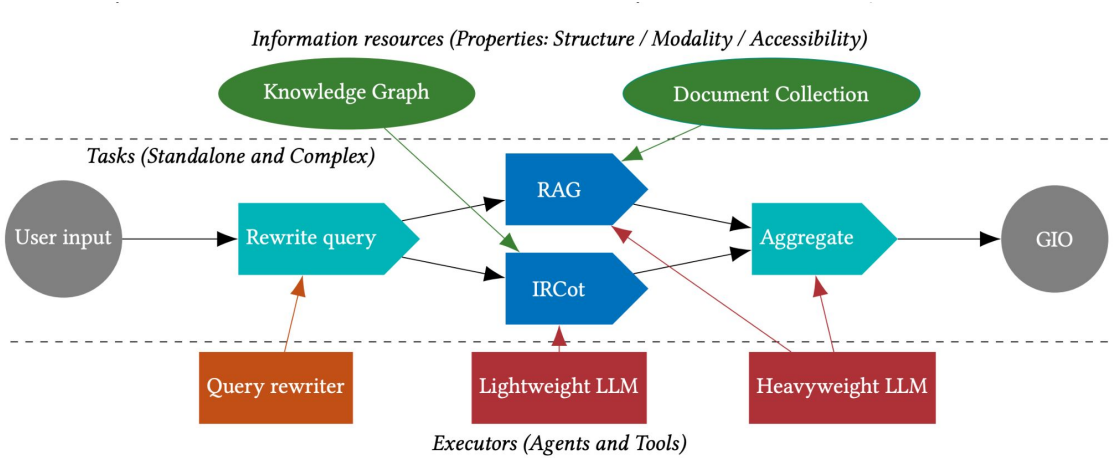
GPTSwarm: Language Agents as Optimizable Graphs. Zhuge et al., 2024.

Multi-Agent Architectures

Adaptive Orchestration



GPTSwarm



AQA

Adaptive Orchestration of Modular Generative Information Access Systems. Hoveyda et al., 2025.

Multi-Agent Architectures

Adaptive Orchestration

Table 3: Performance of each individual agent on the training set (by F1-score and average time in seconds).

	NoR		OneR		IRCoT	
	F1	Time (s)	F1	Time (s)	F1	Time (s)
Context A	0.914	0.66	0.677	6.46	0.730	189.78
Context B	0.061	0.66	0.518	7.34	0.580	192.30
Context C	0.066	0.67	0.146	6.41	0.458	184.85
Overall	0.347	0.66	0.447	6.74	0.589	188.97

Table 4: Evaluation of AQA (NT: Time-agnostic reward, T: Time-based reward) and GPTSwarm [96] on the test set by F1-score and time (log-transformed, in ms). Both have been trained on the training set. For GPTSwarm, the final optimized graph configuration is used for evaluation.

	AQA (NT)		AQA (T)		GPTSwarm	
	F1	Time	F1	Time	F1	Time
Context A	1.0	6.18	1.0	6.18	0.862	12.78
Context B	0.568	12.04	0.539	8.73	0.327	12.79
Context C	0.523	11.75	0.523	11.75	0.317	12.76
Overall	0.697	9.99	0.687	8.89	0.502	12.78

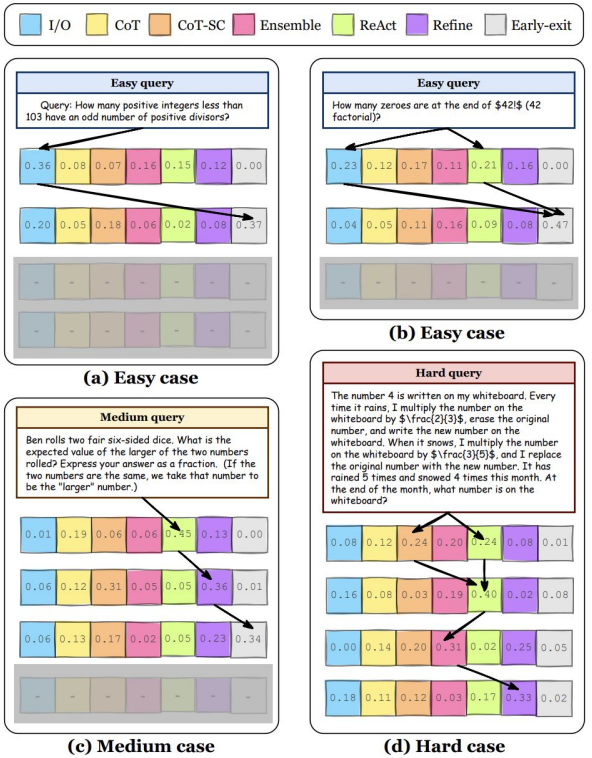
Adaptive Orchestration of Modular Generative Information Access Systems. Hoveyda et al., 2025.

Multi-Agent Architectures

Adaptive Orchestration

Adaptive Orchestration:

Reasoning about what search strategy to execute for a given query.



Multi-agent Architecture Search via Agentic Supernet. Zhang et al., 2025.

Multi-Agent Architectures

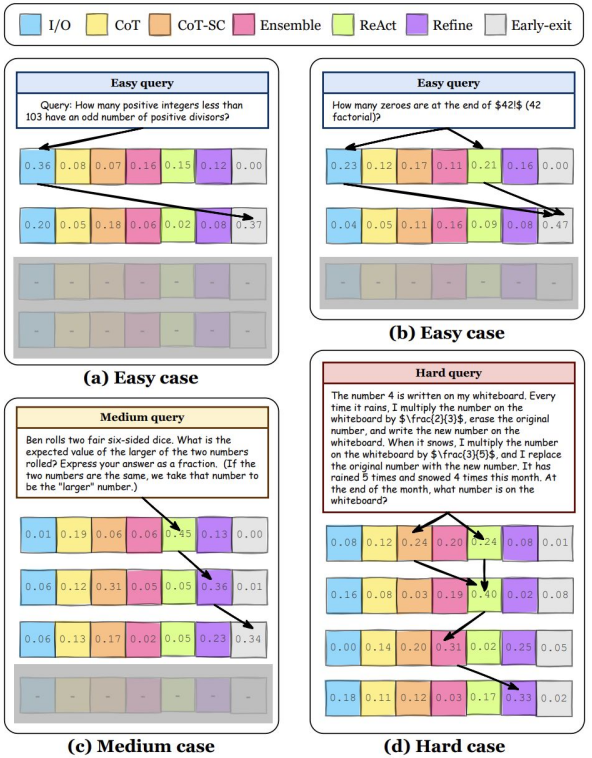
Adaptive Orchestration

Adaptive Orchestration:

Reasoning about what search strategy to execute for a given query.

Open challenges:

1. Dynamic routing mid-trajectory
2. Efficient training
3. Granularity of node decomposition
4. Evaluation



Multi-agent Architecture Search via Agentic Supernet. Zhang et al., 2025.

What will cause multi-agent systems fail?

1. **Error Cascading**
2. **Coordination Overhead & Inter-Agent Misalignment**
   a. Inter-agent communication ⟶ latency and token cost
   b. Redundant reasoning: agents re-derive shared context
   c. More agents ≠ better reasoning past a threshold
   d. Message-passing evaluation protocols needed
3. **Attribution & Verifiability**
   a. Which agent caused the error? Blame attribution is non-trivial
   b. Reasoning trace is distributed across agents ⟶ Inference faithfulness becomes harder.
   c. Current verifier implementations are often insufficient; sole reliance on final-stage

Why Do Multi-Agent LLM Systems Fail? Cemri et al., 2025.

Limitations of current LLM-based approaches,
motivate exploring other approaches;
Neuro-symbolic, Bayesian, geometric, …

Next section: Reasoning methods beyond LLMs

Why Do Multi-Agent LLM Systems Fail? Cemri et al., 2025.

Coffee Break

4. Alternative methods for modeling reasoning

4.1 Neuro-Symbolic Reasoning

Neuro Symbolic Computing

Neuro Symbolic Computing in ML

Neuro-symbolic computing aims at integrating two most fundamental cognitive abilities: the ability to learn from the environment, and the ability to reason from what has been learned. Neuro-symbolic machine learning aims to reconcile the advantages of robust learning in neural networks and reasoning and interpretability of symbolic representation.

E.g., International Conference on Neurosymbolic Learning and Reasoning (NeSy), since 2005.

Neuro Symbolic Computing in IR

The primary application area for neuro symbolic methods in modern IR has been aimed at integrating knowledge graphs in (dense) retrieval.

Disclaimer about ranking with entity annotations

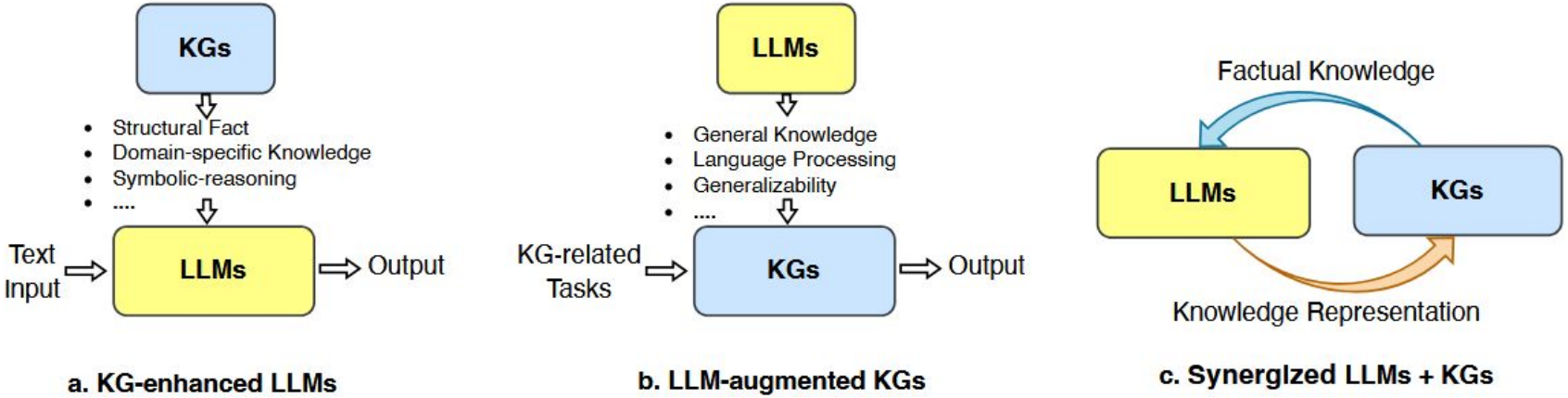
The question whether open domain entity linking (annotating documents and queries with their entity mentions) can really help improve the retrieval system's effectiveness is still an open one. Especially in domain-specific settings, entity annotations are also useful on their own, e.g., to continue browsing from the search results, or to

Characterizing a Semantics for Cognitive Computation

"The aim here is to identify a way of looking at and manipulating commonsense knowledge that is consistent with and can support what we consider to be the two most fundamental aspects of intelligent cognitive behavior: the ability to learn from experience, and the ability to reason from what has been learned. We are therefore seeking a semantics of knowledge that can computationally support the basic phenomena of intelligent behavior."

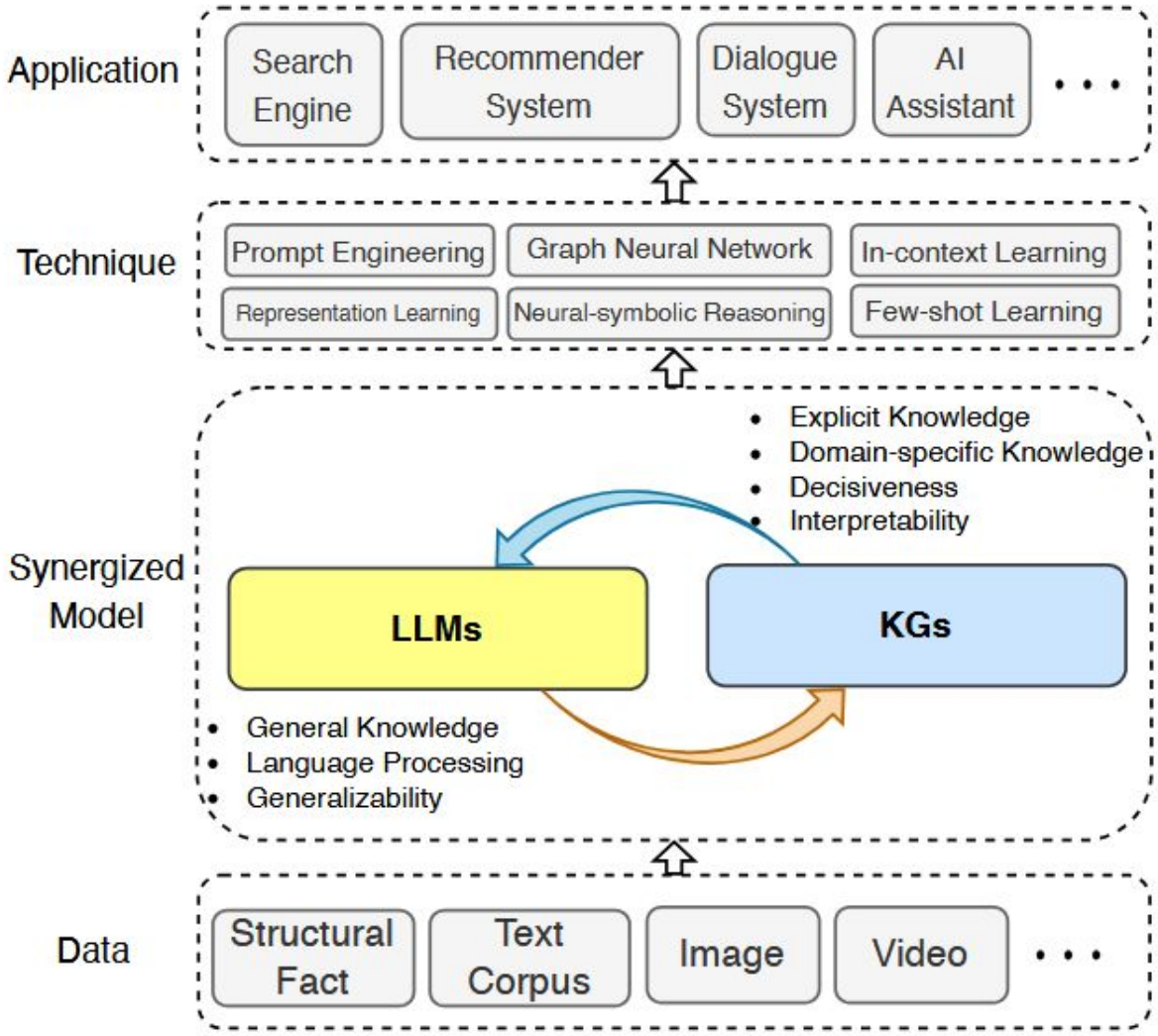
Valiant, 2003

Knowledge Graphs (KGs) and LLMs

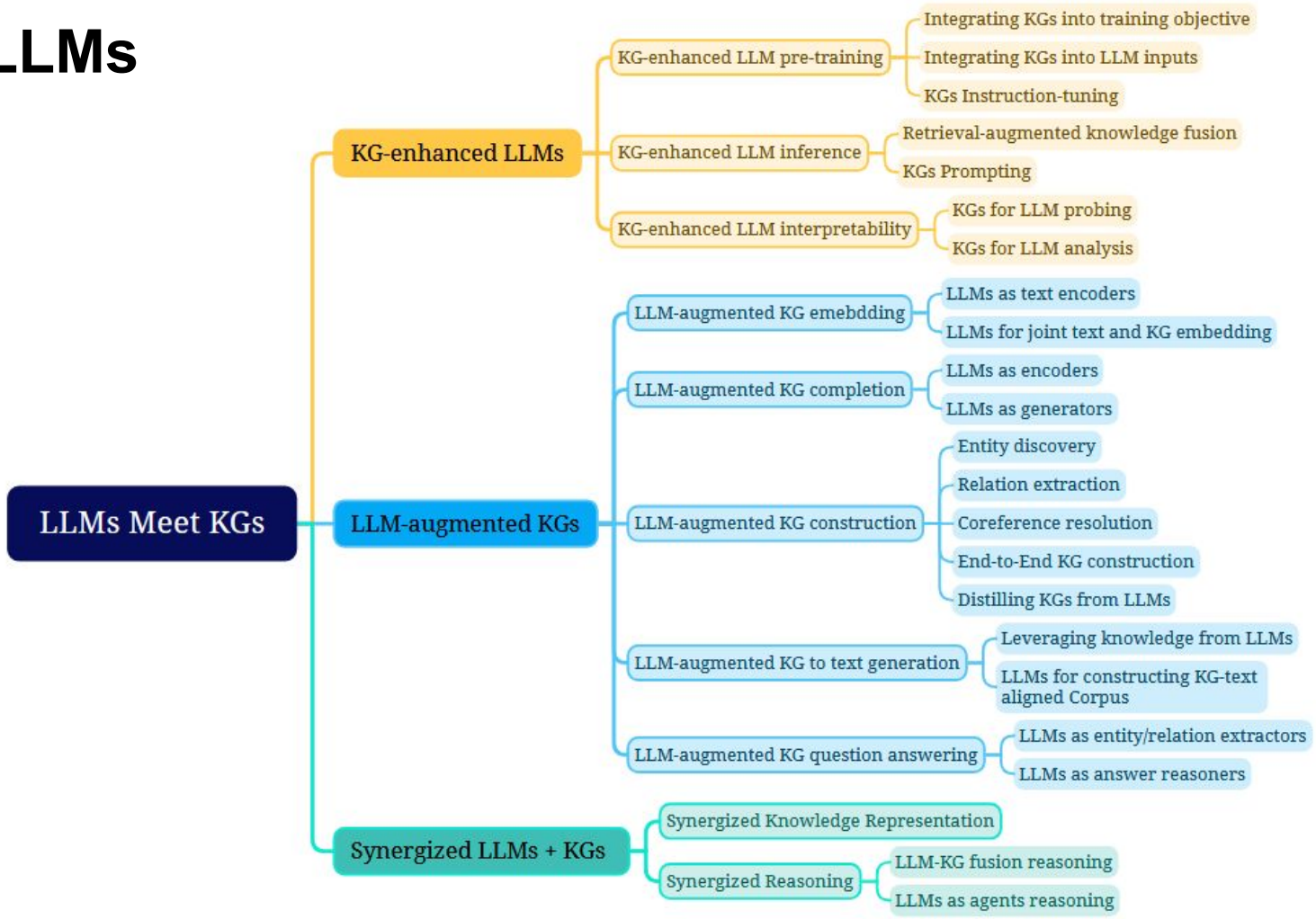


Pan et al., “Unifying Large Language Models and Knowledge Graphs: A Roadmap”, IEEE TKDE 36(7):3580-3599, 2024.

KGs and LLMs


Application
Search Engine
Recommender System
Dialogue System
AI Assistant
Technique
Prompt Engineering
Graph Neural Network
In-context Learning
Representation Learning
Neural-symbolic Reasoning
Few-shot Learning
Synergized Model
Explicit Knowledge
Domain-specific Knowledge
Decisiveness
Interpretability
LLMs
KGs
General Knowledge
Language Processing
Generalizability
Data
Structural Fact
Text Corpus
Image
Video

KGs and LLMs


LLMs Meet KGs
KG-enhanced LLMs
KG-enhanced LLM pre-training
Integrating KGs into training objective
Integrating KGs into LLM inputs
KGs Instruction-tuning
KG-enhanced LLM inference
Retrieval-augmented knowledge fusion
KGs Prompting
KG-enhanced LLM interpretability
KGs for LLM probing
KGs for LLM analysis
LLM-augmented KGs
LLM-augmented KG emebdding
LLMs as text encoders
LLMs for joint text and KG embedding
LLM-augmented KG completion
LLMs as encoders
LLMs as generators
LLM-augmented KG construction
Entity discovery
Relation extraction
Coreference resolution
End-to-End KG construction
Distilling KGs from LLMs
LLM-augmented KG to text generation
Leveraging knowledge from LLMs
LLMs for constructing KG-text aligned Corpus
LLM-augmented KG question answering
LLMs as entity/relation extractors
LLMs as answer reasoners
Synergized LLMs + KGs
Synergized Knowledge Representation
Synergized Reasoning
LLM-KG fusion reasoning
LLMs as agents reasoning

Information Retrieval and Logic

In 1986, two papers by Keith van Rijsbergen initiated quite a movement aiming to develop retrieval models grounded in (non-classical) logic, using **plausible inference** instead of a strict inference.

1. PREFACE

In the last few years, I have become increasingly dissatisfied with the state-of-the-art in information retrieval. I have reluctantly concluded that the fundamental basis of all the previous work is wrong. Almost all of the previous work in Information Retrieval (including my own) has been based on the assumption that a formal notion of meaning is not required to solve the information retrieval problems. Typically, researchers have assumed that one could get by, by only considering absence or presence of word tokens in text together with counting information about the distribution of words. Although such an approach has been successful up to a point, it has become clear that further advances in the effectiveness of retrieval by such techniques are not possible. My observation is that performance based on statistical techniques has reached its theoretical limit and any attempts to achieve further improvements are a waste of time. This is not to say that systems based on these techniques are not worth building; on the contrary, they are because it is the best we have to date. But to build a new generation of Information Retrieval systems, a new theory will be needed.

Information Retrieval and Logic

In 1986, two papers by Keith van Rijsbergen initiated quite a movement aiming to develop retrieval models grounded in (non-classical) logic, using **plausible inference** instead of a strict inference.

An extra benefit of such a new theory for information retrieval would be that its formalization using, e.g., situation theory (Huibers et al., 1996) could also serve as a meta-theory for IR; a theory that could help decide questions as whether a retrieval model should be non-monotonic (i.e., relevance estimates may change upon the availability of new information).

Why logic for IR?

Consider two users with different ideas of relevance but submitting the same request; they should get different probabilities of relevance.

Van Rijsbergen (1986) proposes to separate the logical implication $s \rightarrow q$ from the measure of its uncertainty, $P(s \rightarrow q)$

- separating the process of revising probabilities from the logic, and
- separating the treatment of relevance from the treatment of documents and request.

Why logic for IR?

Chiaramella and Chevallet (1992) argue that:

> “The logic model thus shows in a neat way how knowledge bases could be used while processing queries.”
>
> “This intrinsic power of logic is the best available way for the integration of recent approaches such as AI and NLP within a coherent, controllable, framework.”

Why logic for IR?

Fuhr (2000) introduces a combination of Datalog with probability theory, writing:

> “This approach allows for easy formulation of specific retrieval models for arbitrary applications, and classical probabilistic IR models can be implemented by specifying the appropriate rules. In comparison to other approaches, the possibility of recursive rules allows for **more powerful inferences**, and predicate logic gives the **expressiveness required for multimedia retrieval**. Furthermore, probabilistic Datalog can be used as a query language for **integrated information retrieval and database systems**.”

More grip on “relevance”?

Different relevance criteria:

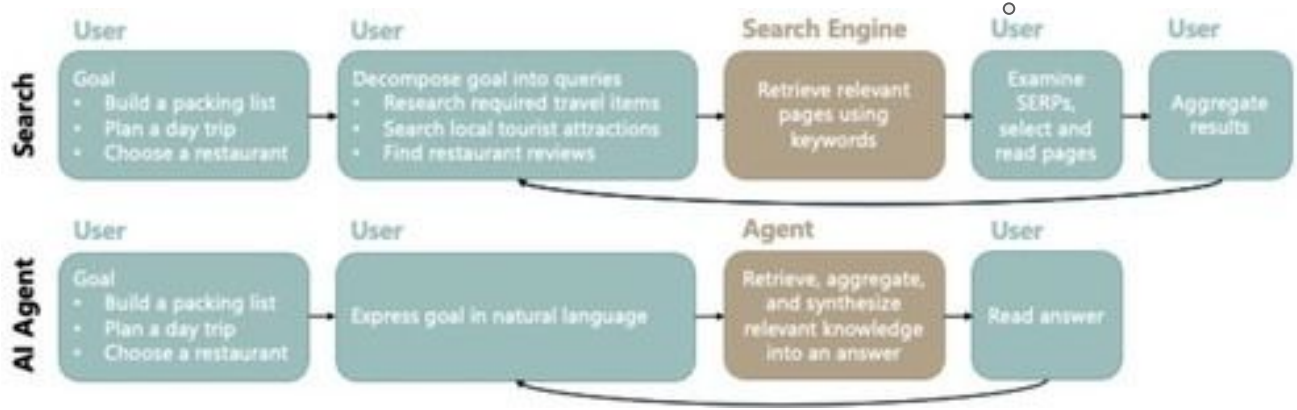
- Topicality: the classic question of **aboutness**
- Readability: can I understand the text?
- Authoritativeness: can I trust the text?
- Child-suitability, timeliness, *etc*

A logical model for IR (e.g., specified as a probabilistic datalog program) would allow us to express the relevance criteria directly in the retrieval model, instead of handling these separately.

More grip on “relevance”?

Really, user goals in natural language only?

Inherently dependent on user, task, context.



With “agentic search” (White, 2024), more of task and context will become part of the “search” than before, not less! Defining the expected result precisely will be crucial input to drive “search agents” and assess success or failure of intermediate steps.

LINC: Logical Inference via Neurosymbolic Computation

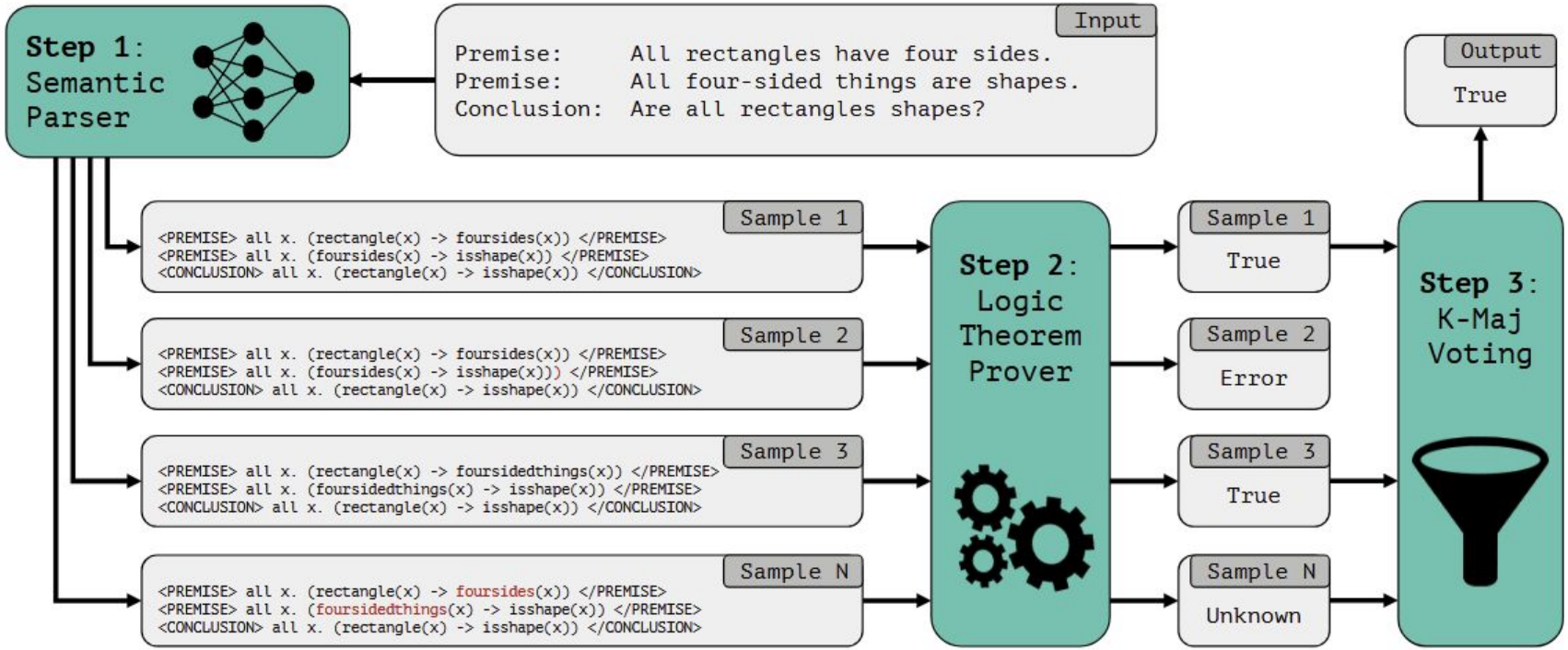
A two-step neurosymbolic process for logical reasoning:

Convert natural language premises and desired conclusion into first-order logic.

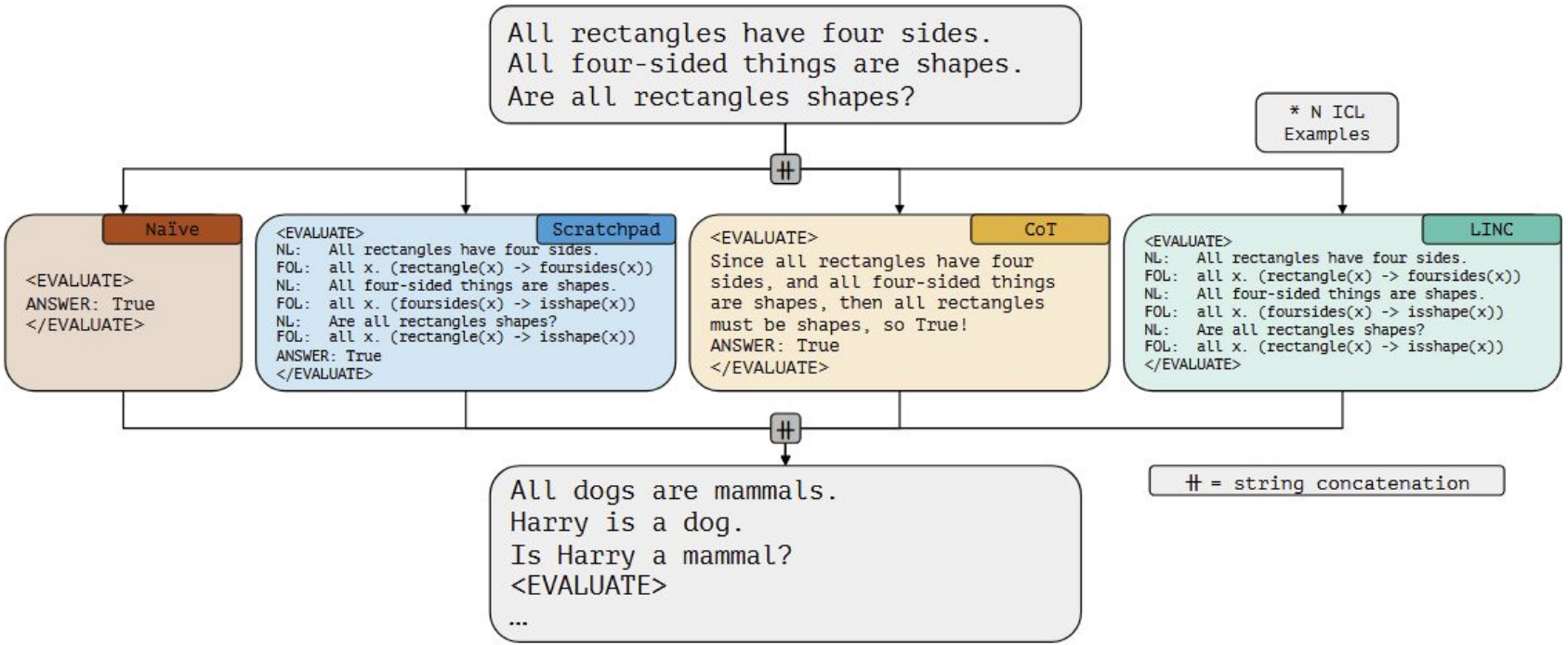
Let a symbolic FOL theorem prover algorithmically determine the truth value of the conclusion given the formalized premises.

In practice, they try 10 different attempted formulations from step 1 and apply a third majority voting step.

LINC: Logical Inference via Neurosymbolic Computation



LINC: Logical Inference via Neurosymbolic Computation



Failure Modes comparison

L1: FOL fails to capture implicit information not mentioned in the premises.

L2: FOL fails to capture information explicitly mentioned in the premises due to the choice of representation.

L3: FOL contains syntax errors.

C1: CoT concludes something different than it suggests.

C2: CoT makes incorrect logical deductions.

C3: CoT fails to find complex paths of reasoning.

LINC and CoT mispredict on different examples, and mispredictions of in-context reasoning baselines are more similar to each other than to LINC.

Theorem Proving for IR?

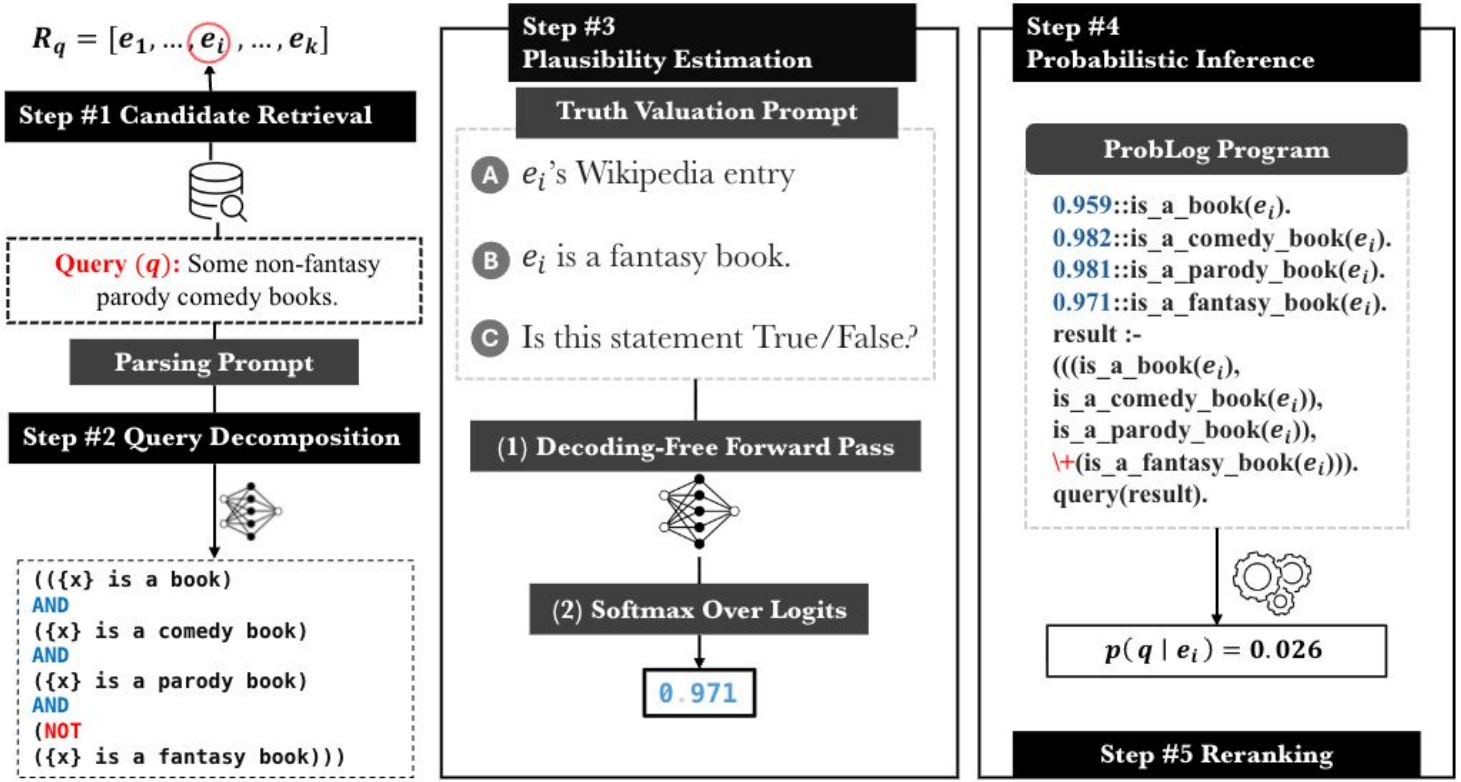
Imagine a **Logician** who knows Probabilistic Prolog.

She answers the complex entity-seeking queries from the QUEST dataset (sets of Wikipedia entities defined through combinations of set-theoretic operations, intersection, union, and difference).

The Logician consults the **Oracle,** an LLM that estimates the plausibility of the facts stated in the ProbLog programme. *Note that the Oracle could be wrong!*

RA CF IF

OrLog



OrLog results

OrLog consistently improves top-rank results and shows its largest advantages on disjunctive structures, where the LLM treats disjunctive constraints as conjunction.

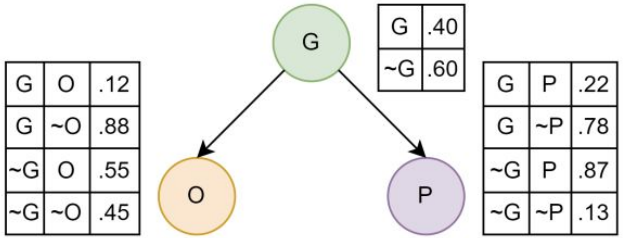
End-to-end LLM reasoning uses 6-10 times more tokens than OrLog:

(i) decoding-free predicate plausibility estimation (a single forward pass per predicate)

(ii) one-off query parsing with composition delegated to ProbLog, whose runtime and computational cost are negligible relative to an LLM call.

Go see Mohanna present the OrLog paper in the plenary, on Tuesday 10.30.

ProbLog vs LLM for Probabilistic Reasoning



Context
G event is True with 40% Probability. P event is True with 22% Probability, if G event is True. P event is True with 87% Probability, if G event is False. O event is True with 12% Probability, if G event is True. O event is True with 55% Probability, if G event is False.

Query	Answer
Extract the probability numbers.	G=0.40, P|G=0.22, ...
Generate the Bayesian Network.	G -> P, G -> O
What is the probability of event P?	61%

Figure 1: An example from the BLInD dataset including an underlying Bayesian network, its textual description, and probabilistic queries in natural language form.

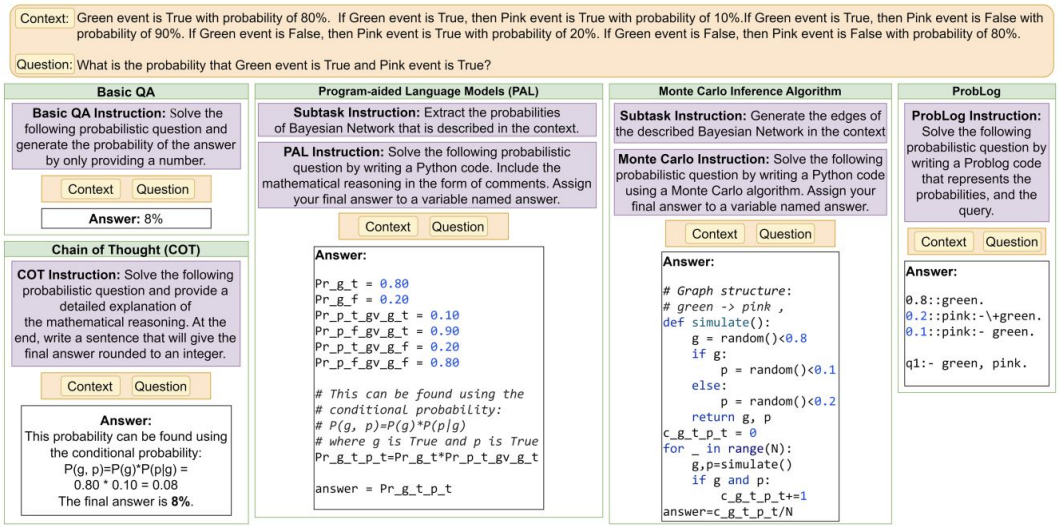


Figure 2: This figure shows our main prompting approaches, PAL, Monte Carlo, and ProbLog, alongside the baseline approaches, Basic QA and COT. Each prompt begins with an instruction (purple boxes) that describes the problem and the answer format. Then, the context, question, and answer are demonstrated depending on the approach. We display only the first in-context example here but use 3 in our experiments. When we require the use of our designed subtasks in the prompt, their instructions and answers are prepended to the main approach, as shown in the PAL method for *Number Extraction* and the Monte Carlo method for *Graph Generation*.

ProbLog vs LLM for Probabilistic Reasoning

Model	Method	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9	V_{10}	V_{2-5}	V_{6-10}	V_{2-10}
GPT3.5	PAL	66	34	25	17	14	9	6	5	2	35	7	19
	PAL w/NE	85	66	41	27	19	12	5	3	6	54	9	29
	Monte Carlo	79	63	71	65	41	32	33	18	14	69	27	46
	Monte Carlo w/GG	85	82	83	68	42	31	28	18	8	79	25	49
	ProbLog	87	82	88	75	59	52	46	38	35	83	46	62
Llama3	PAL	100	84	57	36	31	20	10	14	8	69	17	40
	PAL w/NE	100	95	71	52	46	28	16	16	9	79	23	48
	Monte Carlo	100	100	96	96	92	85	77	72	64	98	78	87
	ProbLog	90	95	92	87	95	94	87	82	78	91	87	89
GPT4	PAL	100	86	70	58	50	27	21	14	7	78	24	48
	PAL w/NE	99	96	78	64	43	26	14	14	10	84	21	49
	Monte Carlo	100	99	98	100	92	94	92	90	88	99	91	94
	Monte Carlo w/GG	100	97	99	98	97	96	88	92	85	99	92	95
	ProbLog	99	98	100	100	96	97	97	98	96	99	97	98

Table 4: GPT3.5, Llama3, and GPT4 accuracy results, presented as percentages, for the PAL, Monte Carlo, and ProbLog methods. w/NE and w/GG denote the inclusion of *Number Extraction* and *Graph Generation*. The columns represent dataset splits V_i, and the average results for smaller BNs V_{2-5}, larger BNs V_{6-10}, and all BNs V_{2-10}.

Wrapping up Part 4.1

Neuro-Symbolic Computing

Reconciles the advantages of robust learning in neural networks and reasoning and interpretability of symbolic representation.

KGs and LLMs

The amount of research that mixes knowledge graphs and large language models is tremendous, but has as yet not had a lot of impact on our field specifically.

Logic and IR

The idea to apply (non-classical) logics to improve information retrieval has a long history in the field. Modelling information access symbolically may have many advantages. Will we manage to finally take up this huge challenge again?

LLMs & Probabilistic Logic Programming

LLMs can turn complex questions into probabilistic logic programs and provide the probability estimates needed for a solver like ProbLog to provide the answer, more accurately and at a lower cost.

4.2. Probabilistic and Bayesian frameworks

Outline

What we have seen so far

- Reasoning as a multi-step inference strategy (e.g., CoT, "thinking")
- Learning-to-reason on static information with grounded signals and gradient-based model updates

What this session discusses

- Bayesian reasoning as a framework to navigate uncertainty
- Updating a model's "beliefs" via observation

Why Bayesian methods show up in LLM work

- LLMs are powerful but overconfident, poorly calibrated and brittle in multi-step reasoning
- The **papers discussed in this section** *(next slide)* introduce
  - explicit beliefs
  - principled updating
  - decision-making under uncertainty

Question: Where do we plug Bayesian methodology into the LLM training?

Papers discussed in this section

Yin et al., **"Reasoning in Flux: Enhancing Large Language Models Reasoning through Uncertainty-aware Adaptive Guidance"**, *ACL* 2024

Tonolini et al., **"Bayesian Prompt Ensembles: Model Uncertainty Estimation for Black-Box Large Language Models"**, *Findings of ACL* 2024

Agarwal et al., **"AutoDiscovery: Open-ended Scientific Discovery via Bayesian Surprise"**, *NeurIPS* 2025

Feng et al., **"BIRD: A Trustworthy Bayesian Inference Framework for Large Language Models"**, *ICLR* 2025

Wang et al., **"BLoB: Bayesian Low-Rank Adaptation by Backpropagation for Large Language Models"**, *NeurIPS* 2024

Hoffman et al., **"Training Chain-of-Thought via Latent-Variable Inference"**, *NeurIPS* 2023

Qiu et al., **"Bayesian teaching enables probabilistic reasoning in large language models"**, *Nature Comms* 2026

Navigate uncertainty

Update beliefs

The unifying lens

- Step 1: Choose the uncertain object
- Step 2: Approximate a posterior over it
- Step 3: Use that posterior to act

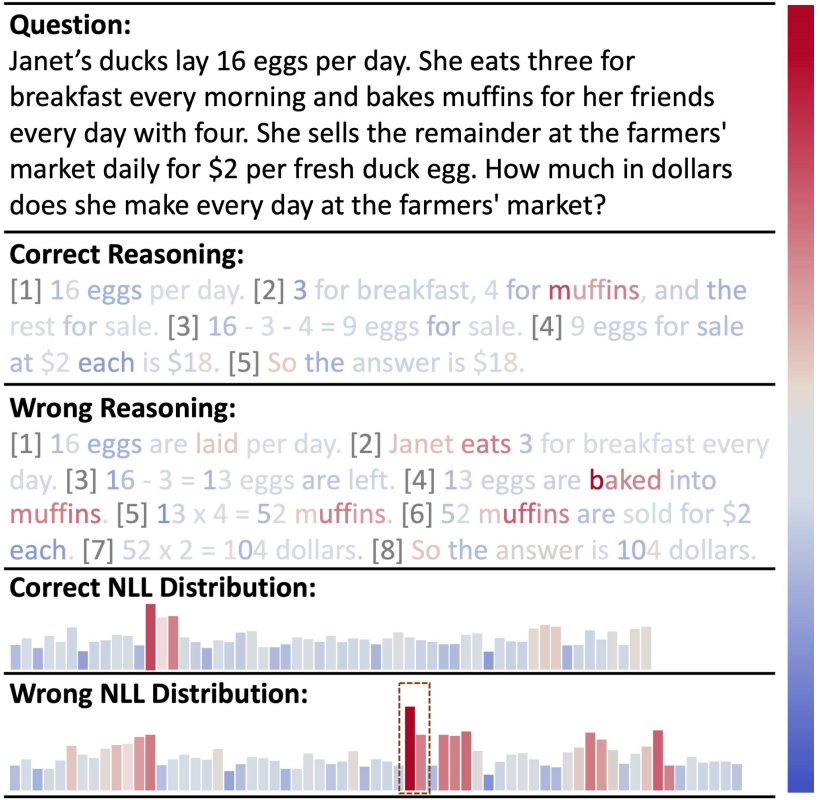
Uncertain objects

- hypothesis support (AutoDiscovery)
- reasoning uncertainty (Reasoning in Flux)
- decision factors (BIRD)
- prompts/instructions (Bayesian PE)
- user preferences (Bayesian teaching)
- rationales (TRICE)
- adapter weights (BLoB)

Bayes as a control signal

Reasoning in Flux (Yin et al., 2024)

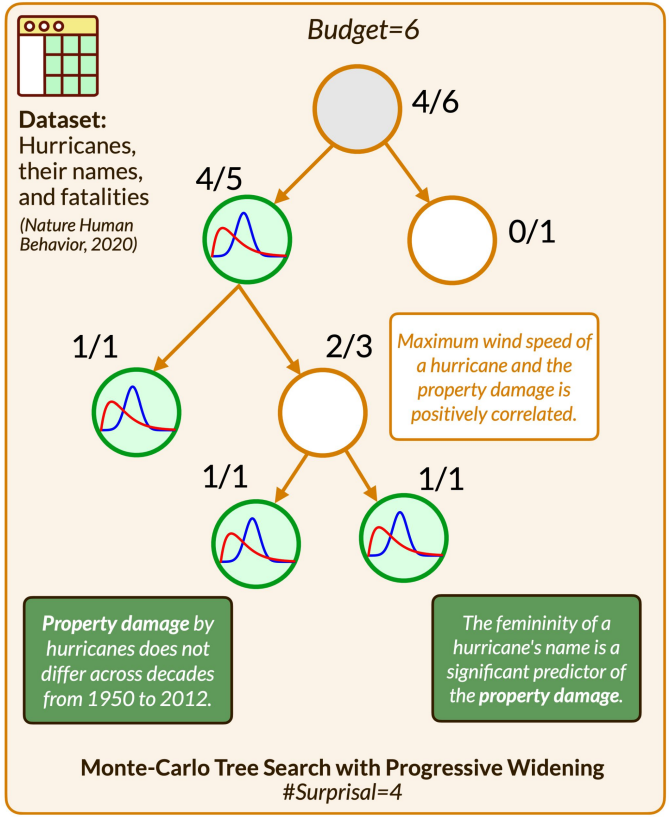
- Tracks token-level uncertainty during CoT
- When uncertainty jumps, it backtracks
- Then injects "reasoning clues" to repair the chain



Bayes as a control signal

AutoDiscovery (Agarwal et al., 2025)

- Models belief in hypothesis support with Beta distributions
- Uses Bayesian surprise as reward
- Plugs that reward into MCTS for open-ended scientific discovery



Bayes as a control signal

Reasoning in Flux (Yin et al., 2024)

- Tracks token-level uncertainty during CoT
- When uncertainty jumps, it backtracks
- Then injects "reasoning clues" to repair the chain

AutoDiscovery (Agarwal et al., 2025)

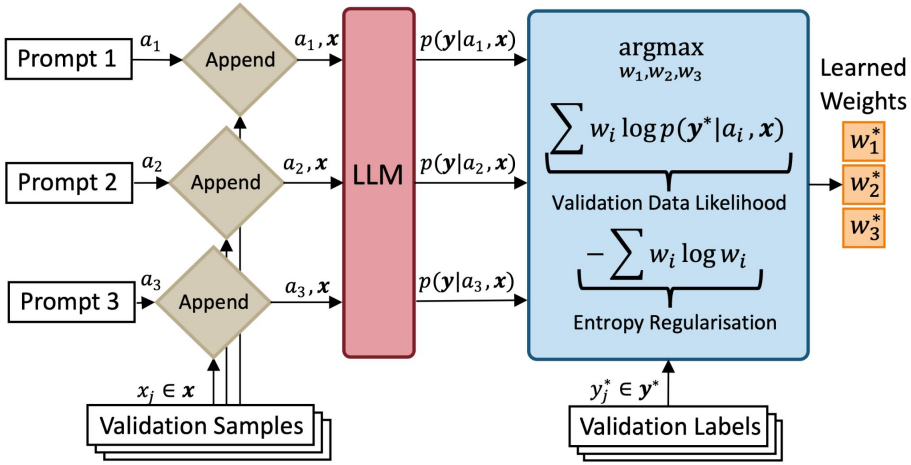
- Models belief in hypothesis support with Beta distributions
- Uses Bayesian surprise as reward
- Plugs that reward into MCTS for open-ended scientific discovery

Common theme: Uncertainty matters, because it changes the next action

Bayes as a black-box enhancer

Bayesian PE (Tonolini et al., 2024)

- Treats prompts as a latent variable
- Builds a weighted prompt ensemble
- Targets black-box LLMs where weight uncertainty quantification is impossible

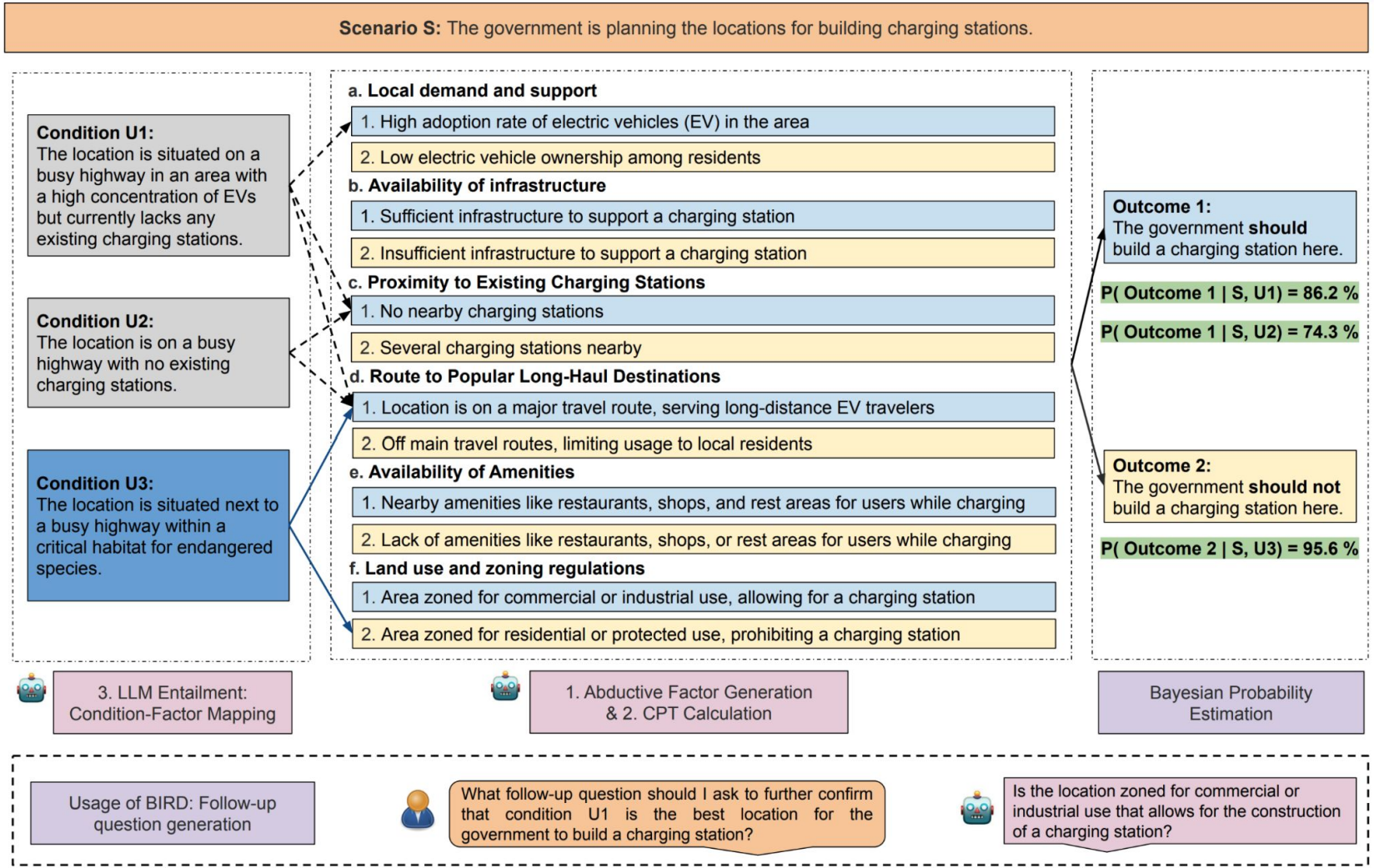


Bayes as a black-box enhancer

BIRD (Feng et al., 2025)

- Keeps LLM as a generator of factors and entailments
- Moves final probability estimation into an external Bayesian model
- Aims for interpretable, controllable decision probabilities

(Figure too *cool* to fit here)


Scenario S: The government is planning the locations for building charging stations.
Condition U1:
The location is situated on a busy highway in an area with a high concentration of EVs but currently lacks any existing charging stations.
Condition U2:
The location is on a busy highway with no existing charging stations.
Condition U3:
The location is situated next to a busy highway within a critical habitat for endangered species.
a. Local demand and support
1. High adoption rate of electric vehicles (EV) in the area
2. Low electric vehicle ownership among residents
b. Availability of infrastructure
1. Sufficient infrastructure to support a charging station
2. Insufficient infrastructure to support a charging station
c. Proximity to Existing Charging Stations
1. No nearby charging stations
2. Several charging stations nearby
d. Route to Popular Long-Haul Destinations
1. Location is on a major travel route, serving long-distance EV travelers
2. Off main travel routes, limiting usage to local residents
e. Availability of Amenities
1. Nearby amenities like restaurants, shops, and rest areas for users while charging
2. Lack of amenities like restaurants, shops, or rest areas for users while charging
f. Land use and zoning regulations
1. Area zoned for commercial or industrial use, allowing for a charging station
2. Area zoned for residential or protected use, prohibiting a charging station
Outcome 1:
The government should build a charging station here.
P(Outcome 1 | S, U1) = 86.2 %
P(Outcome 1 | S, U2) = 74.3 %
Outcome 2:
The government should not build a charging station here.
P(Outcome 2 | S, U3) = 95.6 %
3. LLM Entailment: Condition-Factor Mapping
1. Abductive Factor Generation & 2. CPT Calculation
Bayesian Probability Estimation
Usage of BIRD: Follow-up question generation
What follow-up question should I ask to further confirm that condition U1 is the best location for the government to build a charging station?
Is the location zoned for commercial or industrial use that allows for the construction of a charging station?

Bayes as a black-box enhancer

Bayesian PE (Tonolini et al., 2024)

- Treats prompts as a latent variable
- Builds a weighted prompt ensemble
- Targets black-box LLMs where weight uncertainty quantification is impossible

BIRD (Feng et al., 2025)

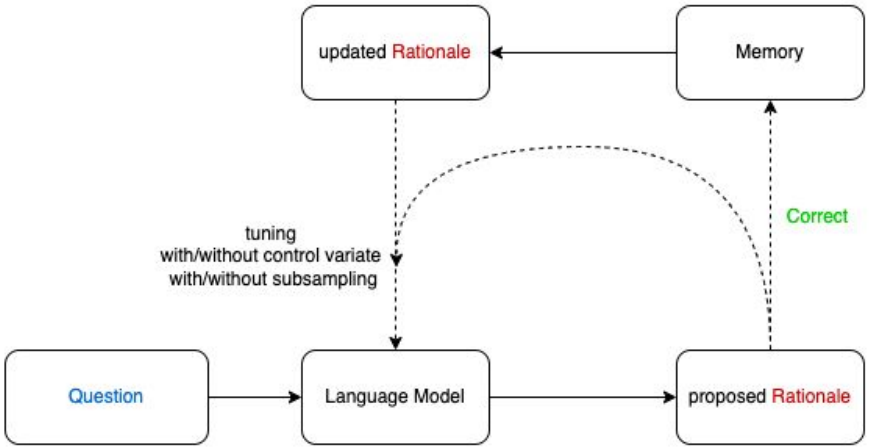
- Keeps LLM as a generator of factors and entailments
- Moves final probability estimation into an external Bayesian model
- Aims for interpretable, controllable decision probabilities

Common theme: Keep the same base model, wrap it in a probabilistic layer

Learning Bayesian behavior inside the model

TRICE (Hoffman et al., 2023)

- Treats CoT as a latent variable
- Trains by maximizing answer likelihood marginalized over rationales
- Avoids need for "gold" rationales

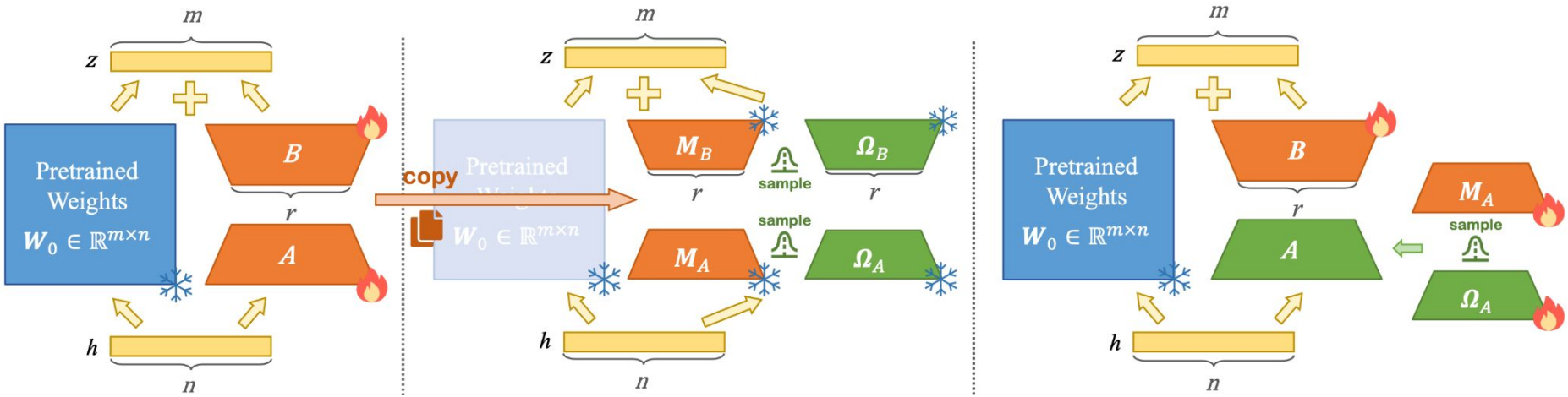


Learning Bayesian behavior inside the model

BLoB (Wang et al., 2024)

- Learns a variational posterior over LoRA adapters during fine-tuning
- Brings Bayesian uncertainty into efficient parameter adaptation

(Figure too *cool* to fit here)

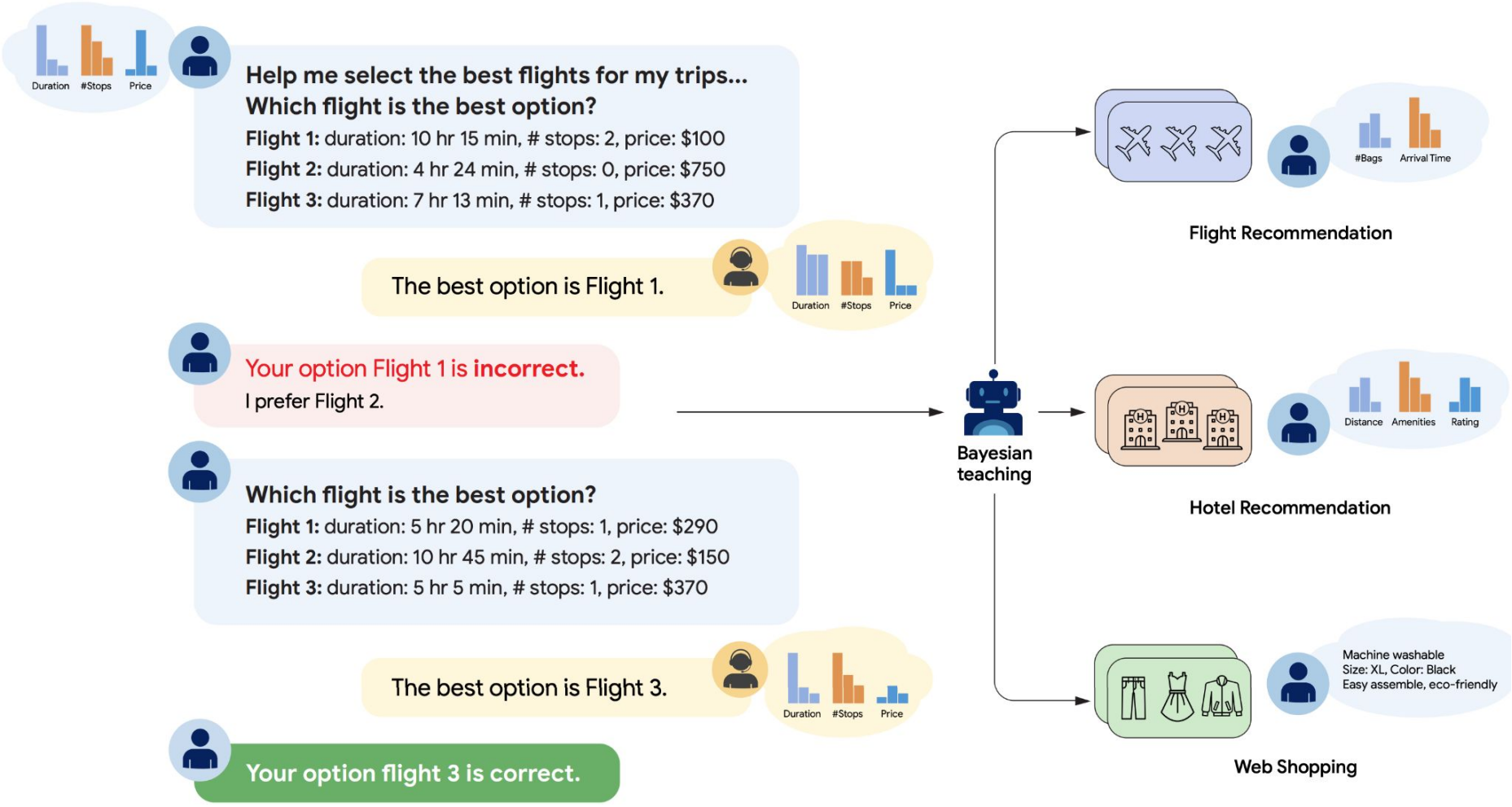

m
z
Pretrained Weights
$W_0 \in \mathbb{R}^{m \times n}$
B
r
A
h
n
copy
Pretrained
$W_0 \in \mathbb{R}^{m \times n}$
M_B
Ω_B
sample
M_A
Ω_A

Learning Bayesian behavior inside the model

Bayesian teaching (Qiu et al., 2026)

- Trains dialogues from a normative Bayesian assistant
- Teaches model to update beliefs over multiple rounds

(Figure too *cool* to fit here)


Duration
#Stops
Price
Help me select the best flights for my trips...
Which flight is the best option?
Flight 1: duration: 10 hr 15 min, # stops: 2, price: $100
Flight 2: duration: 4 hr 24 min, # stops: 0, price: $750
Flight 3: duration: 7 hr 13 min, # stops: 1, price: $370
The best option is Flight 1.
Duration
#Stops
Price
Your option Flight 1 is incorrect.
I prefer Flight 2.
Which flight is the best option?
Flight 1: duration: 5 hr 20 min, # stops: 1, price: $290
Flight 2: duration: 10 hr 45 min, # stops: 2, price: $150
Flight 3: duration: 5 hr 5 min, # stops: 1, price: $370
The best option is Flight 3.
Duration
#Stops
Price
Your option flight 3 is correct.
Bayesian teaching
#Bags
Arrival Time
Flight Recommendation
Distance
Amenities
Rating
Hotel Recommendation
Machine washable
Size: XL, Color: Black
Easy assemble, eco-friendly
Web Shopping

Learning Bayesian behavior inside the model

TRICE (Hoffman et al., 2023)

- Treats CoT as a latent variable
- Trains by maximizing answer likelihood marginalized over rationales
- Avoids need for "gold" rationales

BLoB (Wang et al., 2024)

- Learns a variational posterior over LoRA adapters during fine-tuning
- Brings Bayesian uncertainty into efficient parameter adaptation

Bayesian teaching (Qiu et al., 2026)

- Trains dialogues from a normative Bayesian assistant
- Teaches model to update beliefs over multiple rounds

Common theme: No wrapping; "bake" Bayesian behavior into training

Common themes

- Uncertainty is treated as an engineering primitive
- Each paper chooses a latent variable and uses approximate inference
- Bayesian signals are often used for control, not confidence scoring

Differences

- Token uncertainty vs prompt uncertainty vs parameter uncertainty
- Wrapper methods vs training methods
- Tractability assumptions
  - prompt semantic equivalence
  - factor independence
  - faithful belief elicitation

Takeaways

- Bayesian reasoning with LLMs means
  - choosing what is uncertain
  - approximating a posterior
  - using it to make better decisions
- The papers in this tutorial use Bayesian methods to
  - explore better
  - predict more reliably
  - "repair" reasoning
  - update beliefs
  - fine-tune with calibrated uncertainty

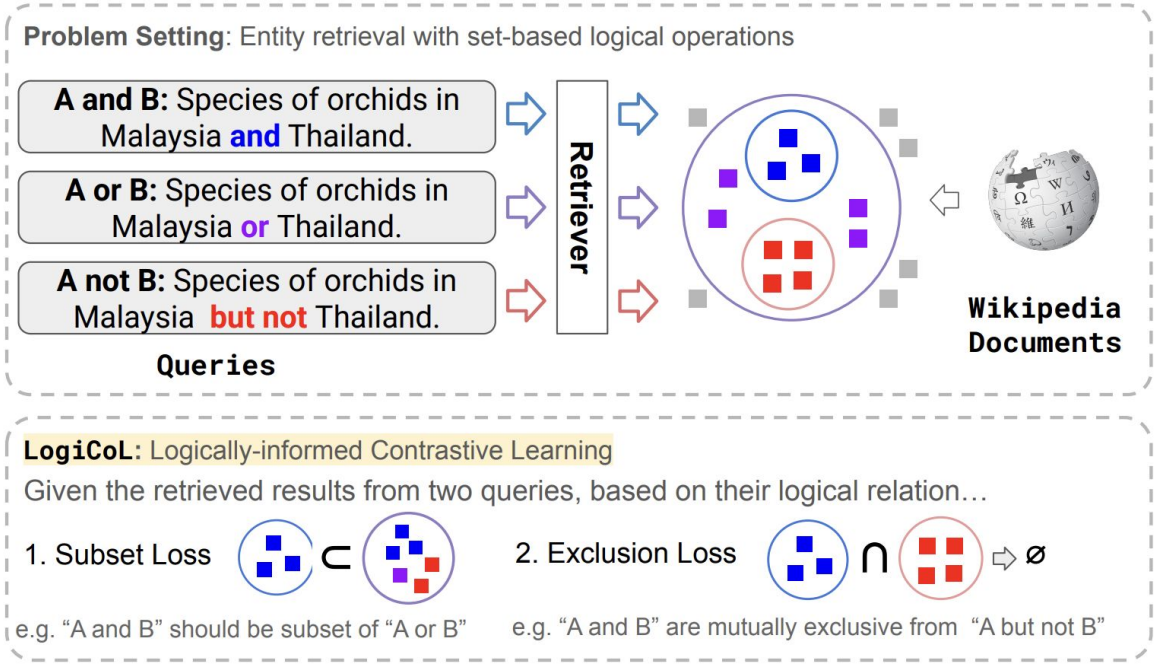
Bottom line

Can Bayesian methods enhance the reasoning capabilities of LLMs?

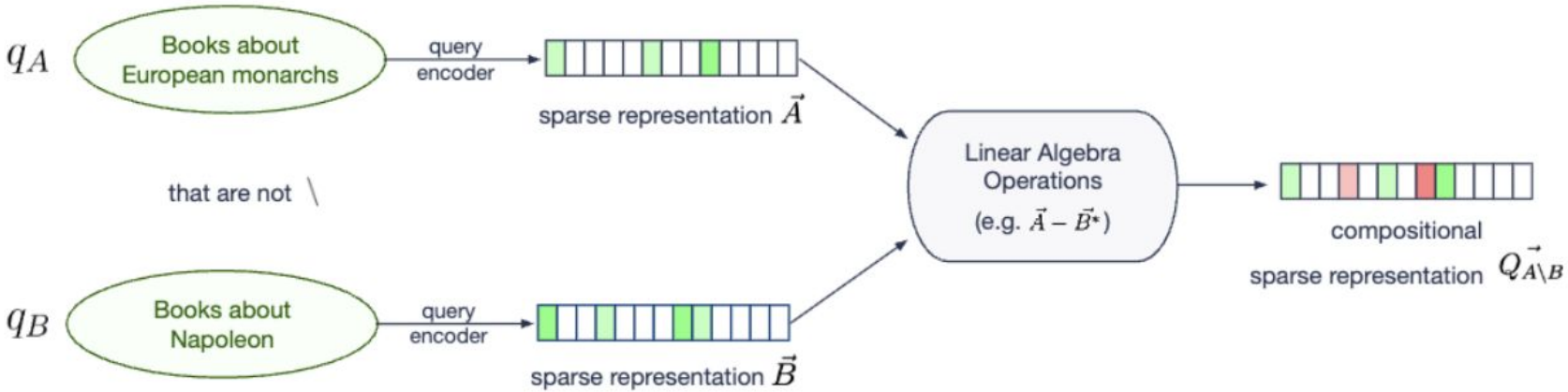
Probably

4.3. Enhancing the representation space

Enhanced Representation for Set-Compositional Queries (LogiCol)

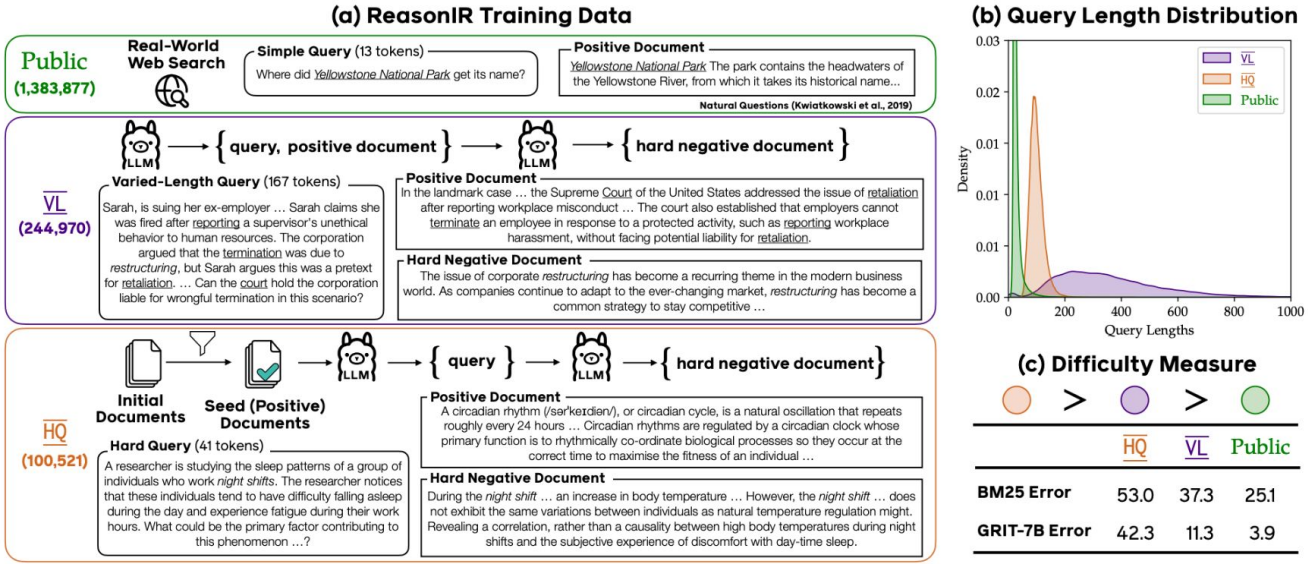


Enhanced Representation for Set-Compositional Queries (SetComp LSR)



4.4. Enhancing the training data

ReasonIR



(b) Query Length Distribution

(c) Difficulty Measure

	HQ	VL	Public
BM25 Error	53.0	37.3	25.1
GRIT-7B Error	42.3	11.3	3.9

5. Bridging current methods/challenges & future directions

Comparative analysis of existing reasoning paradigms along key dimensions:

A. Representational Capability,
B. Inference Faithfulness,
C. Computational Feasibility.

Here again: A 12-type taxonomy, with evaluation coverage

Reasoning type	IR manifestation	Coverage
Deductive	Logical entailment between query and document	Partial
Inductive	Generalizing from retrieved evidence to claims	Partial
Abductive	Query intent inference; hypothesis formation from evidence	Untested
Analogical	Cross-domain retrieval; structural relevance beyond surface	Untested
Causal	Distinguishing correlation from intervention evidence	Untested
Defeasible	Provisional relevance; belief revision on new retrieval	Untested
Modal	Tracking claim strength (established/contested/speculative)	Untested
Epistemic	Knowledge state tracking across a literature over time	Untested
Dialectical	Mapping argumentative position within a field's disputes	Untested
Temporal	Evidence currency; supersession; historical state tracking	Partial
Metacognitive	Query reformulation; knowing when to stop; uncertainty flag	Partial
Practical	Search strategy decisions in agentic pipelines	Untested

How do the discussed methods compare?

Representational capability

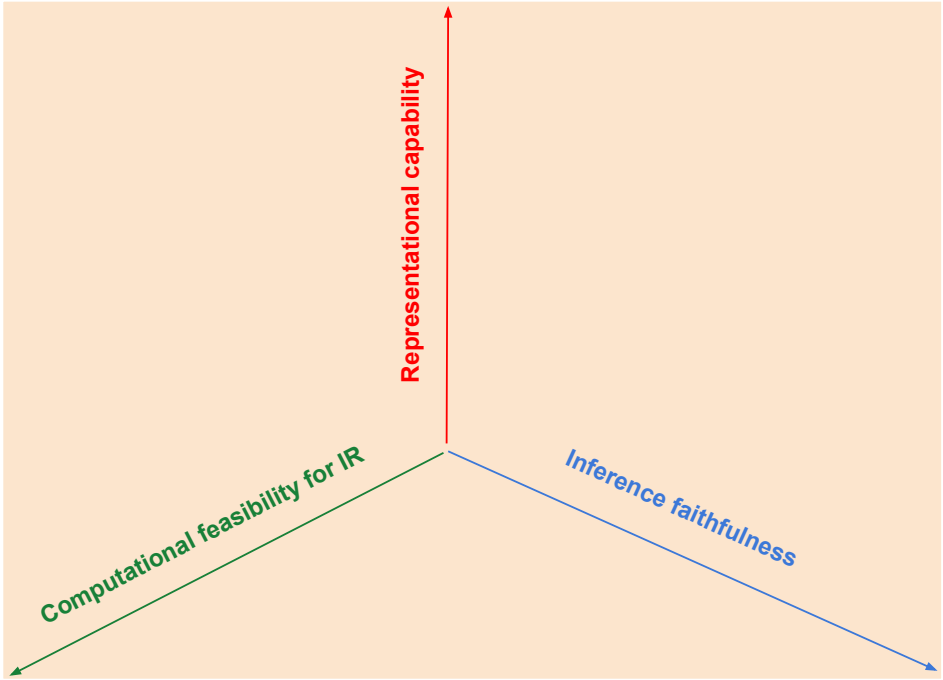
E.g., Does reasoning proposals explicitly encode uncertainty, causal relationships, etc.?

Inference faithfulness

Can faithfulness of the reasoning process be verified?

Computational feasibility for IR

Is the reasoning process computationally feasible over a large set of documents?



Where would you place the methods?

Representational capability

E.g., Does reasoning proposals explicitly encode uncertainty, causal relationships, etc.?

Inference faithfulness

Can faithfulness of the reasoning process be verified?

Computational feasibility for IR

Is the reasoning process computationally feasible over a large set of documents?

Inference-time reasoning with LLMs

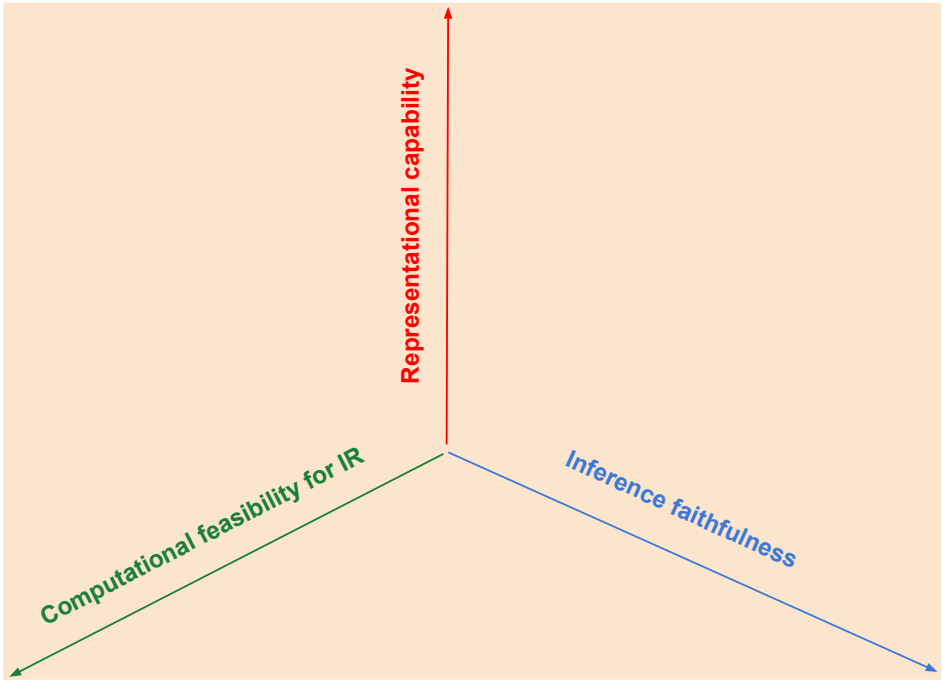
RL-training LLMs for reasoning

Neuro-symbolic

Bayesian Reasoning

Enhancing the representation space

Enhancing training data



Open questions?

References

Section 2

1) Alhamoud, K. et al., 2025. NegBench: Vision-language models do not understand negation. CVPR.
2) Bender, E., et al. 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? FAccT.
3) Chen et al., 2025. Reasoning Models Don't Always Say What They Think. arXiv.
4) Chen, Z., et al., 2025. Browsecomp-plus: A more fair and transparent evaluation benchmark of deep-research agent. arXiv.
5) DeepSeek, 2025. DeepSeek-R1: Incentivizing Reasoning Capability in LLMs via Reinforcement Learning. arXiv.
6) Dziri, N., et al., 2023. Faith and Fate: Limits of Transformers on Compositionality. NeurIPS.
7) Fodor, J., 2025. Line Goes Up? Inherent Limitations of Benchmarks for Evaluating Large Language Models. arXiv.
8) Kalai, A.T., et al., 2025. Why language models hallucinate. arXiv.
9) Lightman, et al., 2023. Let's Verify Step by Step. ICLR.
10) Liu, et al. 2024. Lost in the Middle: How Language Models Use Long Contexts. TACL
11) Madabushi H., et al., 2025. Neither Stochastic Parroting nor AGI: LLMs Solve Tasks through Context-Directed Extrapolation from Training Data Priors.
12) Malaviya, C., et al., 2023. Quest: A retrieval dataset of entity-seeking queries with implicit set operations. ACL.
13) Mirzadeh, et al., 2025. GSM-Symbolic: Understanding the Limitations of Mathematical Reasoning in Large Language Models. ICLR.
14) OpenAI 2024. Learning to Reason with LLMs. (OpenAI o1 system card)
15) Petcu, R., et al., 2025. A comprehensive taxonomy of negation for NLP and neural retrievers. Findings of EMNLP.
16) Ravichandat, A., et al., 2022. CONDAQA: A contrastive reading comprehension dataset for reasoning about negation. EMNLP.
17) Schaeffer et al., 2023. Are Emergent Abilities of Large Language Models a Mirage. NeurIPS.
18) Shi F., et al., 2023. Large Language Models Can be Easily Distracted by Irrelevant Context. ICML.
19) Su, H.,et al., 2025. BRIGHT: A realistic and challenging benchmark for reasoning-intensive retrieval. ICLR.
20) van den Elsen, C., et al., 2025. Reproducing NevIR: Negation in neural information retrieval. SIGIR.
21) Wang et al., 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. ICLR.
22) Wang, X., et al., 2024. BIRCO: A benchmark of information retrieval tasks with complex objectives. arXiv.
23) Wei et al., 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. NeurIPS.
24) Weller, O., et al., 2024. NevIR: Negation in neural information retrieval. EACL.
25) Weller, O., et al., 2025. FollowIR: Evaluating and teaching information retrieval models to follow instructions. NAACL.
26) Weller, O., et al., 2026. On the theoretical limitations of embedding-based retrieval. ICLR.
27) Yao et al., 2023. Tree of Thoughts: Deliberate Problem Solving with Large Language Models. NeurIPS.
28) Zhang, H., et al., 2024. A Careful Examination of Large Language Model Performance on Grade School Arithmetic (GSM1K). arXiv.
29) Zhang, W., et al., 2024. ExcluIR: Exclusionary neural information retrieval. AAAI.
30) Zhou et al., 2023. Least-to-Most Prompting Enables Complex Reasoning in Large Language Models. ICLR.

Section 3

1) Merrill & Sabharwal. The Parallelism Tradeoff: Limitations of Log-Precision Transformers. TACL 2023.
2) Feng et al. Towards Revealing the Mystery behind Chain of Thought: A Theoretical Perspective. NeurIPS 2023.
3) Wei et al. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. NIPS 2022.
4) Ruis et al. Procedural Knowledge in Pretraining Drives Reasoning in Large Language Models. ICLR 2025.
5) Wang & Zhou. Chain-of-Thought Reasoning without Prompting. NeurIPS 2024.
6) Madaan, et al. Self-refine: Iterative refinement with self-feedback. NeurIPS 2023.
7) Yao et al. React: Synergizing reasoning and acting in language models. ICLR 2023.
8) Hoveyda et al., Adaptive Orchestration of Modular Generative Information Access Systems. SIGIR 2025.
9) Training language models to follow instructions with human feedback. Ouyang et al., NeurIPS 2022.
10) DeepSeekMath: Pushing the Limits of Mathematical Reasoning in Open Language Models. Shao et al., 2024.
11) DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. Guo et al., 2025.
12) Search-R1: Training LLMs to Reason and Leverage Search Engines with Reinforcement Learning. Jin et al., COLM 2025.
13) Rank-R1: Enhancing Reasoning in LLM-based Document Rerankers via Reinforcement Learning. Zhuang et al., 2025.
14) Does Reinforcement Learning Really Incentivize Reasoning Capacity in LLMs Beyond the Base Model? Yue et al., NeurIPS 2025.
15) GPTSwarm: Language Agents as Optimizable Graphs. Zhuge et al., ICML 2024.
16) Adaptive Orchestration of Modular Generative Information Access Systems. Hoveyda et al., SIGIR 2025.
17) Multi-agent Architecture Search via Agentic Supernet. Zhang et al., ICML 2025.
18) Why Do Multi-Agent LLM Systems Fail? Cemri et al., 2025.

Section 4

Neuro-Symbolic:

1) A. d'Avila Garcez et al., “Neural-Symbolic Computing: An Effective Methodology for Principled Integration of Machine Learning and Reasoning”, FLAP 6(4): 611-632 (2019)
2) L.G. Valiant, “Three problems in computer science.” J. ACM, 50(1):96–99, 2003.
3) Pan et al., “Unifying Large Language Models and Knowledge Graphs: A Roadmap”, IEEE TKDE 36(7):3580-3599, 2024.
4) C.J. van Rijsbergen, “A new theoretical framework for information retrieval”, in SIGIR ‘86.
5) C.J. van Rijsbergen, “A non-classical logic for information retrieval”, The Computer Journal, 29(6):481-485, 1986.
6) C.J. van Rijsbergen, “Towards an information logic”, in SIGIR ‘89.
7) Y. Chiaramella and J.P. Chevallet, “About Retrieval Models and Logic”, The Computer Journal, 35(3), 1992.
8) T.W.C. Huibers, M. Lalmas and C.J. van Rijsbergen, “Information Retrieval and Situation Theory”, SIGIR Forum 30(1):11-25, 1996.
9) N. Fuhr, “Probabilistic datalog: Implementing logical information retrieval for advanced applications”, J. Am. Soc. Inf. Sci., 51: 95-110, 2000.
10) S. Mizzaro, “How many relevances in information retrieval?”, Interacting with Computers, 10(3):303-320, 1998.
11) Olausson et al., “ LINC: A neuro-symbolic approach for logical reasoning by combining language models with first-order logic provers”, in EMNLP 2023.
12) Hoveyda et al., “OrLog: Resolving Complex Queries with LLMs and Probabilistic Reasoning”, in ECIR 2026.
13) Nafar et al., “Reasoning over Uncertain Text by Generative Large Language Models”, in AAAI 2025.
14) R.W. White, “Advancing the Search Frontier with AI Agents”. Commun. ACM 67(9):54–65, 2024.

Section 4

Bayesian:

1) Hoffman et al., "Training Chain-of-Thought via Latent-Variable Inference", NeurIPS 2023
2) Tonolini et al., "Bayesian Prompt Ensembles: Model Uncertainty Estimation for Black-Box Large Language Models", Findings of ACL 2024
3) Yin et al., "Reasoning in Flux: Enhancing Large Language Models Reasoning through Uncertainty-aware Adaptive Guidance", ACL 2024
4) Wang et al., "BLoB: Bayesian Low-Rank Adaptation by Backpropagation for Large Language Models", NeurIPS 2024
5) Agarwal et al., "AutoDiscovery: Open-ended Scientific Discovery via Bayesian Surprise", NeurIPS 2025
6) Feng et al., "BIRD: A Trustworthy Bayesian Inference Framework for Large Language Models", ICLR 2025
7) Qiu et al., "Bayesian teaching enables probabilistic reasoning in large language models", Nature Comms 2026

Remaining:

8) LogiCoL: Logically-Informed Contrastive Learning for Set-based Dense Retrieval. Shen et al., ACL 2025.
9) Constructing Set-Compositional and Negated Representations for First-Stage Ranking. Krasakis et al., CIKM 2025.
10) ReasonIR: Training Retrievers for Reasoning Tasks. Shao et al., 2025.