Superficial Reflection or Genuine Thought? A Fine-Grained Cognitive Analysis of Large Reasoning Models

Yuxiang Chen[1,2*] **Zuohan Wu**[3*] **Ziwei Wang**[3] **Xiangning Yu**[4] **Xujia Li**[5†]
Linyi Yang[6] **Mengyue Yang**[7] **Jun Wang**[1] **Lei Chen**[3,5]
[1]University College London [2]AI Lab, Yangtze River Delta [3]HKUST (Guangzhou)
[4]Tianjin University [5]HKUST
[6]Southern University of Science and Technology [7]University of Bristol
leexujia@ust.hk

Abstract

Motivated by the observed human-like behaviours in Large Reasoning Models (LRMs), this paper introduces a comprehensive taxonomy to characterise atomic reasoning steps and analyse the reasoning behaviours of LRMs. Grounded in human cognitive processes, we propose a taxonomy comprising five groups and seventeen categories. Through this taxonomy, we conduct an in-depth analysis of contemporary LRMs and distil four actionable takeaways for model optimisation. Most notably, we reveal that prevailing post-answer "double-checks" are largely superficial and rarely yield substantive revisions. A targeted intervention further shows that explicitly eliciting richer reflection processes can substantially improve failed self-correction. To support this large-scale study, we propose CAPO, an automated annotation method used to construct a dataset of 277,534 reasoning steps with strong agreement with human expert annotations. We further validate the main behavioural patterns on a newer reasoning model and a coding domain, demonstrating the broader applicability of the proposed taxonomy. All source code and data are available at https://github.com/hehepig4/psyche.

1 Introduction

Large Reasoning Models (LRMs), such as DeepSeek-R1 (DeepSeek-AI et al., 2025) and OpenAI o1 (Jaech et al., 2024), have demonstrated strong performance on complex tasks such as mathematics and coding. Unlike conventional language models, LRMs often produce extended reasoning traces before giving a final answer, commonly referred to as long chain-of-thought (CoT) (Wei et al., 2022). Interestingly, these traces frequently contain human-like expressions such as "wait", "let me check", or "maybe not", together with behaviours resembling recall, hypothesis generation, reconsideration, and self-checking. Such observations have motivated increasing interest in understanding LRM reasoning from perspectives inspired by human cognition.

Existing analyses, however, remain relatively coarse-grained. A common perspective interprets LRM reasoning through high-level cognitive analogies such as System 1 and System 2 reasoning (Wang, 2025). While useful for conceptualising extended reasoning, such distinctions offer limited resolution for characterising the diverse behaviours within a long CoT. For example, "reflection" may refer to merely checking a previous result, identifying the cause of an error, considering alternatives, or actually changing the reasoning strategy. Treating these processes as a single behaviour can obscure important differences between successful and unsuccessful reasoning.

Meanwhile, CoT outputs should not be interpreted as direct simulations of human neural or cognitive mechanisms. Human cognition emerges from biologically grounded processes, whereas LRMs acquire their behaviours through large-scale data-driven training and reinforcement learning (DeepSeek-AI et al., 2025; Wang et al., 2024a; Feng et al., 2023). We therefore use human cognitive processes as *descriptive templates* for observable reasoning behaviours rather than claims about

[*]Equal contribution.
[†]Corresponding author.

Findings of the Association for Computational Linguistics: IJCNLP-AACL 2026, pages 4530–4548
November 6-10, 2026 ©2026 Association for Computational Linguistics

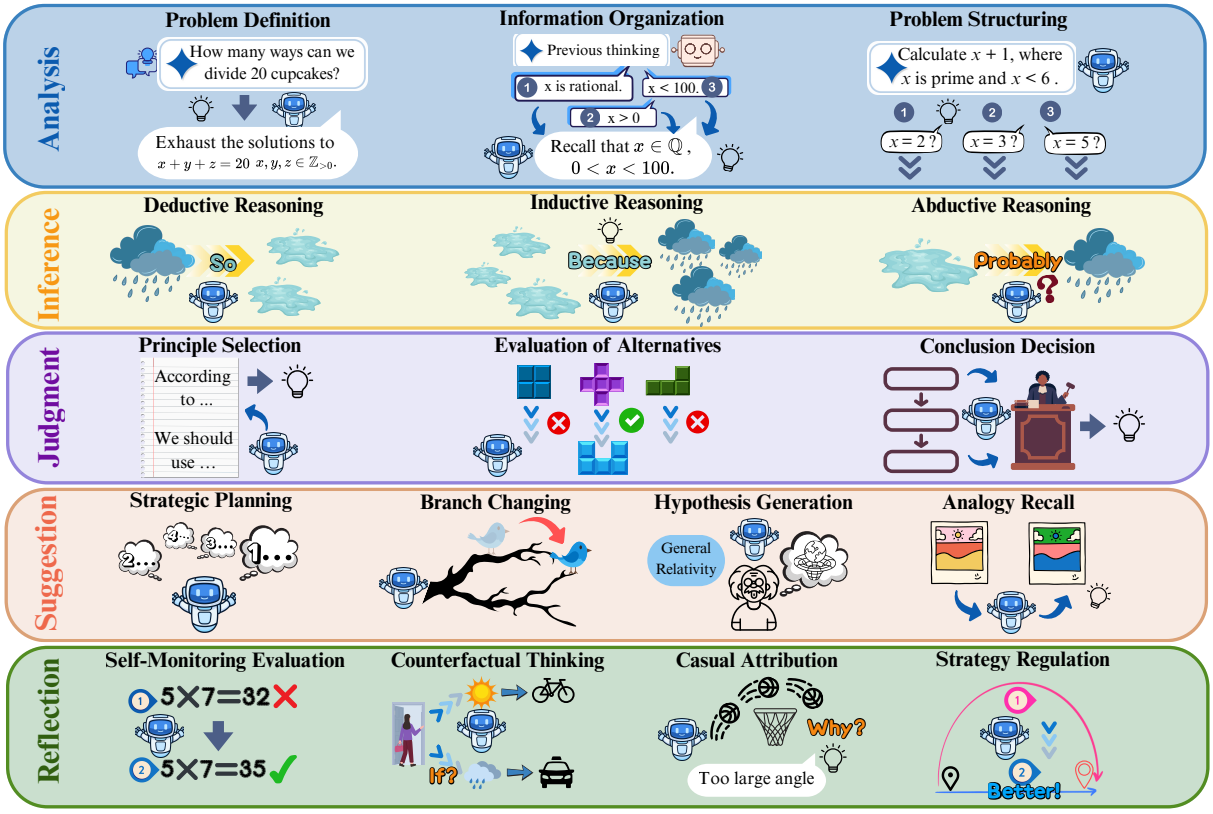


Figure 1: A brief illustration of the proposed taxonomy from human cognitive perspectives.

the internal mechanisms of LRMs.

Motivated by this perspective, we introduce a fine-grained taxonomy for analysing individual steps in LRM reasoning traces. Drawing inspiration from formal logic (Gentzen, 1935; Prawitz, 1965), education theory (Vygotsky, 1978; Wood et al., 1976; Collins et al., 1989), and cognitive psychology (Flavell, 1979; Chi et al., 1989; Dunlosky et al., 2013), the taxonomy contains **five high-level groups and seventeen fine-grained categories**. This hierarchical design preserves commonality at the group level while distinguishing processes that may play very different roles during reasoning, such as self-monitoring, causal attribution, hypothesis generation, and information organisation.

The fine granularity of this taxonomy makes large-scale manual annotation prohibitively expensive. We therefore propose **Constrained Automatic Prompt Optimisation (CAPO)**, which aligns an LLM annotator with expert labels through iterative prompt optimisation. CAPO achieves a micro Cohen's κ of **0.811** and a macro κ of **0.594** against expert annotations, while the principal behavioural signals underlying our analysis are also preserved on the human-annotated subset. Using this framework, we construct a corpus of **277,534 annotated mathematical reasoning steps**, including **9,841 human annotations** and **267,693 CAPO annotations**.

Our analysis reveals four recurring patterns in current LRMs:

1. **Information organisation.** Successful CoTs more frequently recall and reorganise important intermediate information, helping maintain relevant context over long reasoning trajectories.
2. **Analogy & hypothesis.** Hypothesis generation and analogy recall tend to occur later and with weaker justification in unsuccessful reasoning trajectories.
3. **Reflection.** Native post-answer "double-checks" are predominantly shallow self-monitoring and rarely produce substantive revisions. In a targeted intervention, explicitly eliciting richer reflection corrects **43 of 62 (69.3%)** previously failed self-correction cases.
4. **Redundancy.** Long reasoning traces contain substantial numbers of steps with low estimated necessity, suggesting that longer reasoning does not necessarily imply more effective reasoning.

We further examine the framework on the newer MiniMax-M2 model and on CodeForces reasoning traces. Several core behavioural patterns transfer across model generations and reasoning domains, while the taxonomy also captures meaning-

ful changes in newer LRMs. Detailed robustness analyses are reported in Appendix F.

In summary, our contributions are:

1. We introduce a five-group, seventeen-category taxonomy that provides a fine-grained descriptive framework for analysing LRM reasoning trajectories.
2. We propose CAPO for scalable step-level annotation and validate its agreement with expert annotations.
3. We construct a large annotated reasoning corpus and identify four empirical patterns concerning information organisation, speculative reasoning, reflection, and redundancy, complemented by targeted intervention and cross-model and cross-domain validation.

2 Related Work

With the remarkable success of reasoning models such as OpenAI's o1, research has intensified in this area. Following (Chen et al., 2025b), two key behaviour groups are often distinguished. **Deep reasoning**, covering step-wise inference and planning, has been strengthened via prompt engineering (e.g., chain-of-thought) (Wei et al., 2022), specialised decoding/control structures (Besta et al., 2024; Yao et al., 2023), and targeted training (Zelikman et al., 2022). **Reflection** constitutes a second core behaviour group (Chen et al., 2025b), where methods introduce feedback-then-refinement cycles to revise drafts or plans (Tan et al., 2025; Wang et al., 2024b; Kumar et al., 2025). Beyond "how to do" reasoning, recent studies probe where improvement opportunities lie, examining factors such as problem complexity (Shojaee et al., 2025), inference-time scaling (Wu et al., 2024a), error types (He et al., 2025), and confidence (Yoon et al., 2025). However, existing definitions and taxonomies of CoTs remain limited in coverage. A broader, step-wise perspective in an interdisciplinary human lens remains underexplored.

To date, one study in this line first (Marjanovic et al., 2025) proposes a rudimentary four-category taxonomy, reporting preliminary findings as a minor component of a larger agenda. Substantially richer structure nonetheless appears attainable. The idea of classifying thought has a long lineage: from Plato's *Statesman* on classification in governance and knowledge (Plato, 1892) and Aristotle's *Categories* on distinctions underlying reasoning (Aristotle, 1928), to modern accounts in logic (proof theory and natural deduction) (Gentzen, 1935; Prawitz, 1965), education theory (scaffolding and cognitive apprenticeship) (Vygotsky, 1978; Wood et al., 1976; Collins et al., 1989), and cognitive psychology (metacognition and self-explanation) (Flavell, 1979; Chi et al., 1989; Dunlosky et al., 2013). Building on these traditions, this paper advances a fine-grained, step-wise CoT taxonomy and a large-scale annotation methodology (CAPO), enabling comprehensive behavioural analysis beyond coarse stage-based schemes.

3 A CoT Taxonomy through a Human Cognitive Lens

We develop a principled taxonomy for analysing the steps of CoT reasoning produced by LRMs. Recall that CoT sequences are not direct simulations of neural or cognitive mechanisms. Thus, we ground our taxonomy in the pedagogical tradition, treating human thought patterns as expert templates that serve as descriptive lenses for classifying the expression of reasoning in LLMs. Subsequently, we draw inspiration from multiple disciplines that have long examined the nature and structure of human reasoning. As depicted in Figure 1, we refine five high-level categories into seventeen finer-grained subcategories. This hierarchical structure facilitates clear distinctions between reasoning types while retaining their commonality at the higher level. For concrete illustrations of each subcategory, please refer to Appendix A.

Operationalising reasoning steps. We operationalise a reasoning step using the natural paragraph boundaries ("\n\n") produced by the LRMs studied in this work. These boundaries typically correspond to coherent "thought blocks" in long CoTs. Empirically, **82.2%** of the resulting segments receive exactly one mental-process label, while only a small fraction contain more than two labels. We further conduct a robustness analysis on 100 randomly sampled CoTs using sentence-level segmentation. The principal behavioural patterns remain broadly consistent, while sentence-level splitting increases the number of annotation units by approximately **1.63**×. These observations suggest that paragraph-level segmentation provides a practical balance between semantic coherence, analytical granularity, and annotation cost.

Analysis. Analysis entails decomposing complex ideas into constituent parts to understand their interrelations. This phase corresponds to model behaviours where abstract tasks are broken down

into subtasks, either explicitly through prompting or implicitly via attention (Mayer, 1998). From a process-level perspective, this involves:

- **Problem Definition (A.PD)**: Clarifying the problem's core challenge by paraphrasing the question, identifying its type, or surfacing hidden goals and constraints.
- **Problem Structuring (A.PS)**: Breaking the problem into logical parts or subgoals, often by outlining a solution plan or isolating knowns and unknowns.
- **Information Organisation (A.IO)**: Reviewing or restructuring prior information, such as results or premises, to support upcoming reasoning (Punia et al., 2023).

In LLMs, analysis functions as a stabilising scaffold, enabling the model to reason with awareness of context, dependencies, and previously computed results (Punia et al., 2023). Although such steps may not directly advance the final answer, they can play a metacognitive role in maintaining coherence and preserving relevant information throughout long-form reasoning.

Inference. From formal logic and philosophy of science, we incorporate canonical paradigms of inference, including deduction, induction, and abduction (Carnap and Jeffrey, 1971; Johnson-Laird and Byrne, 1991). These forms of inference constitute the backbone of reasoning processes and correspond to different ways in which assertions are drawn from premises, whether necessarily (deduction), probabilistically (induction), or plausibly (abduction):

- **Deductive Reasoning (I.DR)**: Applying general rules to derive logically certain conclusions. This is common in tasks such as mathematical proofs, where valid premises guarantee correct outcomes (Carnap and Jeffrey, 1971).
- **Inductive Reasoning (I.IR)**: Generalising patterns from specific examples. This is typical in empirical reasoning or analogical settings, where conclusions are probable but not certain (O'Rourke and Josephson, 1997).
- **Abductive Reasoning (I.AR)**: Inferring the most plausible explanation for an observation. Although uncertain, it is important for hypothesis generation and commonsense reasoning (Johnson-Laird and Byrne, 1991).

For instance, deductive structures can be observed in arithmetic CoT tasks, inductive patterns emerge in classification or analogy tasks, and abductive moves often appear when the model speculates about hidden causes or intentions. As emphasised in (Dewey, 1910), inference operates as the connective tissue between intermediate reasoning steps, enabling a model to construct coherent lines of argumentation or problem-solving trajectories.

Judgment. Judgment involves comparing alternatives and selecting solutions based on principled reasoning, aligning with pedagogical models of critical thinking, argument evaluation, and decision-making. Within human reasoning theory, judgment typically involves weighing competing hypotheses, assessing consistency with prior knowledge, and selecting the most justified course of action. In LLMs, judgment refers to the evaluative process by which an agent compares alternative solution paths and determines which is most appropriate based on prior reasoning, including three subtypes:

- **Principle Selection (J.PS)**: Choosing appropriate logical principles, ethical rules, or task-specific criteria to guide decision-making.
- **Evaluation of Alternatives (J.EA)**: Comparing competing reasoning paths or hypotheses to select the most viable direction.
- **Conclusion Decision (J.CD)**: Making a final commitment to an answer or solution, justified by prior reasoning steps.

For disambiguation with the Suggestion type introduced next, judgments are conclusions drawn from the preceding reasoning process, whereas suggestions introduce potential directions or information before such evaluation.

Suggestion. Suggestion is informed by studies on spontaneous idea generation, creativity, and the psychological phenomenon of suggestion itself (Gheorghiu et al., 1987), which highlight how new directions in thought can emerge without immediate justification.

In our taxonomy, suggestion refers to the generative act of proposing new ideas that extend beyond the direct content of the problem. It encompasses heuristic and forward-looking reasoning behaviours that introduce novel directions, potential solution paths, or speculative constructs before any evaluation takes place.

- **Strategic Planning (S.SP)**: Proposing a plan or high-level roadmap for how the problem might be approached. This often appears as a declarative intention to structure upcoming reasoning steps (e.g., "First, I will try a substitution, then check

for symmetry").

- **Branch Changing (S.BC)**: Initiating a shift from the current reasoning path to an alternative one, typically when the existing direction is perceived as unproductive.
- **Hypothesis Generation (S.HG)**: Formulating a tentative explanation or educated guess based on limited evidence. It is relevant to tasks involving hidden rules, causality, or implicit goals, where the next step is unclear.
- **Analogy Recall (S.AR)**: Recalling a past experience, familiar structure, or well-known problem to inform the current task. Analogical suggestion often acts as a conceptual scaffold, helping the model bridge from known solutions to new domains.

Understanding and identifying Suggestion steps within CoT sequences thus provides insight into how models initiate, branch, and explore within complex problem spaces.

Reflection. Reflection provides a cognitive model for meta-level reasoning and error awareness (Dewey, 1910). It represents a metacognitive capability to step outside the current stream of reasoning and critically evaluate its validity, necessity, and efficiency:

- **Self-Monitoring Evaluation (R.SME)**: Reviewing the reasoning process so far and checking for errors or inconsistencies in logic.
- **Counterfactual Thinking (R.CT)**: Considering alternative actions or decisions and reasoning about what might have occurred under different conditions. This can be used to reassess current reasoning or outcomes through "what-if" scenarios.
- **Causal Attribution (R.CA)**: Analysing the reasons behind success or failure by identifying the factors or decisions that contributed to the result.
- **Strategy Regulation (R.SR)**: Adjusting the current overall reasoning or problem-solving strategy based on feedback or prior reflection.

Notably, existing studies such as (Qin et al., 2024) have attributed reflection as an important capability underlying successful reasoning in LRMs. Reflection allows backward examination of generated steps, identification of potential errors, and reconsideration of whether the current direction remains appropriate. As we show later, however, different forms of reflection exhibit substantially different behavioural patterns, motivating the need to distinguish self-monitoring from deeper processes such as causal attribution and strategy regulation.

Why fine-grained categories? The seventeen subcategories are intended as a descriptive analytical resolution rather than a claim that they form the unique decomposition of machine reasoning. To examine whether this finer granularity provides additional empirical information, we also aggregate the labels into the five parent categories and repeat our analysis. The coarser representation substantially obscures category-specific patterns: for example, the aggregate Suggestion category exhibits no significant overall difference between successful and unsuccessful reasoning, despite distinct behaviours of Hypothesis Generation and Analogy Recall revealed by the fine-grained analysis. This suggests that the finer taxonomy preserves behavioural signals that can be diluted when heterogeneous processes are merged.

4 CAPO: Scalable LLM-Assisted Annotation

Given the prohibitive cost of manual annotation—for example, 931 AIME CoTs contain 177,687 reasoning steps—and the demonstrated capability of LLMs for data labelling (Chiang and Lee, 2023; Calderon et al., 2025), we leverage LLMs to automate the assignment of our taxonomy categories.

The major challenge lies in aligning LLM annotations with human expert standards (Calderon et al., 2025). Conventional techniques such as *fine-tuning* (FT) and *in-context learning* (ICL) are less suitable for our setting. FT requires relatively large amounts of labelled data, whereas ICL becomes expensive for long reasoning trajectories. Since classifying an individual step may depend on its preceding reasoning context, incorporating multiple full-context demonstrations can substantially increase prompt length and exacerbate long-context limitations, including the *lost-in-the-middle* phenomenon (Liu et al., 2024). To address these issues, we propose **Constrained Automatic Prompt Optimisation (CAPO)**.

Inspired by iterative human learning, CAPO optimises the annotation prompt by analysing discrepancies between zero-shot predictions and expert annotations. We formulate this process within a genetic algorithm (GA) framework comprising three operators: (1) **Mutation**, where the LLM summarises annotation "tips" from individual alignment errors and incorporates them into the prompt; (2) **Reproduction**, where a meta-process synthe-

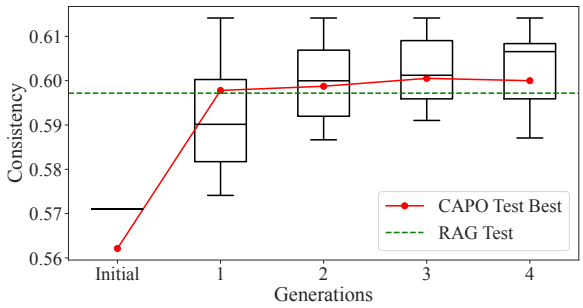


Figure 2: Annotation consistency of CAPO and the RAG baseline. The boxes show the consistency distribution of candidate prompts on the training population, while the red line denotes the test consistency of the best validation-selected prompt.

sises an improved prompt by combining two high-performing prompt variants; and (3) **Elimination**, where candidate prompts are evaluated and underperforming variants are discarded to enable iterative improvement.

To mitigate overfitting to the small expert-labelled training set, we further introduce a *tripartite prompting strategy* that differs from existing GA-based prompt optimisation methods (Guo et al., 2024; Sécheresse et al., 2025; Wu et al., 2024b). Specifically, we partition the prompt into three regions: the **constant region**, which contains invariant task formats; the **variable region**, which contains taxonomy descriptions with only constrained modifications; and the **mutable region**, which remains open-ended and stores task-specific annotation guidance. This design preserves the core taxonomy definitions while allowing CAPO to learn flexible annotation strategies from observed human–model discrepancies. Representative annotation tips learned by CAPO are provided in Appendix D.

4.1 CAPO Evaluation

We evaluate CAPO using *Gemini-2.5-Flash* against a Retrieval-Augmented Generation (RAG) baseline (Marjanovic et al., 2025), which performs ICL with retrieved expert-labelled exemplars. The evaluation is conducted on a split of our human-annotated AIME⋆ and HMMT data.

As shown in Figure 2, CAPO surpasses the RAG baseline after a single optimisation round while avoiding the substantially longer contexts required by retrieved ICL demonstrations. These results show that prompt optimisation provides an effective and comparatively context-efficient way to scale our taxonomy annotation.

Process	Human interval	Machine MD	Within interval
A.IO	$[-0.067, +0.094]$	$+0.011$	✓
S.HG	$[-0.048, +0.018]$	-0.041	✓
S.AR	$[-0.061, +0.008]$	-0.030	✓
R.SME	$[-0.073, +0.092]$	-0.018	✓

Table 1: Comparison of the principal behavioural effects estimated from human and CAPO annotations. MD denotes the mean difference between correct and incorrect CoTs. Full category-level results are provided in the appendix.

Agreement with expert annotations. Because CAPO provides the majority of annotations used in our subsequent analysis, we additionally quantify human–machine agreement using Cohen's κ. CAPO achieves a **micro κ of 0.811** and a **macro κ of 0.594**. The lower macro agreement is mainly concentrated in rare categories, for which substantially fewer expert-labelled examples are available.

More importantly, we separately examine whether the behavioural signals used in our main findings are preserved between the human- and CAPO-annotated subsets. Table 1 reports the four categories most directly involved in the subsequent takeaways. In each case, the machine-estimated mean difference falls within the corresponding interval estimated from human annotations.

Therefore, although CAPO does not reproduce every expert annotation perfectly, the principal signals underlying our subsequent analysis are also supported by the expert-labelled subset. Agreement is lower for several rare reflection categories, such as R.CT and R.CA; we discuss this limitation and provide the full category-level comparison in Appendix D.

5 Mental-Process Patterns in LRM Reasoning

With the taxonomy and annotation framework established, we now analyse real-world LRM reasoning trajectories to identify mental-process patterns associated with successful and unsuccessful reasoning.

Dataset. Our core analysis focuses on mathematical problem solving. For human annotation, we collect thirty problems from the 2025 AIME⋆ (OpenCompass, 2025) and HMMT (Balunović et al., 2025), covering major areas of high-school mathematics. To scale the analysis, we further include MATH (Hendrycks et al., 2021) and AIME (Di Zhang, 2025), which are annotated using CAPO as introduced in Section 4. Unless otherwise specified, the reasoning CoTs are

Name	Correct	Incorrect	Steps
MATH	963 (96.3%)	37 (3.7%)	90,006
AIME	810 (87.0%)	121 (13.0%)	177,687
ComS	1000		7,710
HMMT	9 (64.3%)	5 (35.7%)	4,375
AIME*	6 (37.5%)	10 (62.5%)	5,466
ComS*	30		254

Table 2: Number of annotated CoTs and reasoning steps. The top three rows use CAPO annotations, whereas the bottom three use human annotations. The core mathematical analysis excludes the two ComS rows.

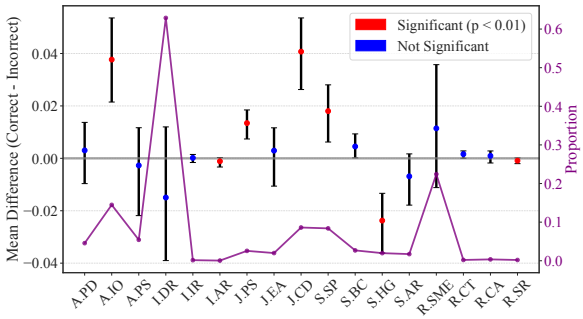


Figure 3: Proportional differences between correct and incorrect CoTs. Red dots indicate statistically significant differences. The purple line denotes the overall proportion of each mental process.

generated by DeepSeek-R1 (0120) (DeepSeek-AI et al., 2025), a representative open-source reasoning model with 671B parameters.

The core mathematical dataset contains **277,534 annotated reasoning steps**, including **9,841 expert annotations** and **267,693 CAPO annotations**. We additionally collect a commonsense subset for qualitative analysis. Dataset statistics are summarised in Table 2, and links are provided in Appendix C.

For the commonsense subset, DeepSeek-R1 answers almost all questions correctly. Because the analyses below explicitly compare correct and incorrect CoTs, we exclude this subset from the corresponding correctness-based analyses, as discussed in Appendix B.

Annotation protocol. Following practices in (He et al., 2025), we segment long CoTs using the natural paragraph delimiter '\n\n'. Annotators identify **all** applicable process tags for **each** reasoning step, making annotation a multi-label classification task. Human annotators receive training and discussion sessions based on the taxonomy in Section 3; large-scale annotations are produced using CAPO. We further examine the robustness of the segmentation choice in Section F.

Based on these trajectories, we investigate which mental-process patterns are associated with successful reasoning. We compress each CoT into a 17-dimensional feature vector in $[0, 1]^{17}$, where each dimension denotes the proportion of one mental process. For each category, we test whether its proportion differs significantly between correct and incorrect CoTs. Figure 3 summarises the results.

Recalling forgotten. LRMs frequently revisit previously established information through Analysis.Information Organisation (A.IO). In long reasoning trajectories, periodically restating intermediate milestones may help keep important information available to subsequent reasoning and mitigate the well-known *lost-in-the-middle* phenomenon (Liu et al., 2024).

Figure 3 shows that the proportion of A.IO is more than 0.04 lower in incorrect CoTs than in correct ones. The following example illustrates a failure in which the model states two relevant conditions but subsequently loses track of the second:

> **52:** *To have G in the last word, two conditions hold:*
> **53:** *1. G is a first letter.*
> **54:** *2. All other first letters are less than G.*
> **...** **Step 53 discussed, step 54 forgotten**.

Takeaway 1

Information organisation is strongly associated with successful reasoning. This motivates training or post-training strategies that encourage models to preserve, recall, and reorganise important intermediate information throughout long reasoning trajectories.

Specious intuitions. Suggestion.Hypothesis Generation (S.HG) and Suggestion.Analogy Recall (S.AR) introduce candidate ideas that are not necessarily accompanied by immediate rigorous derivation. As shown in Figure 3, these processes occur more frequently in failed reasoning trajectories. One possible interpretation is that models become more likely to invoke speculative directions after earlier attempts fail.

We further examine *when* these processes occur. Table 3 reports their average relative positions, where each step position is normalised to $[0, 1]$ by CoT length. Both S.HG and S.AR occur significantly later in incorrect trajectories. This does not imply that analogy or hypothesis generation is inherently harmful; rather, *late and insufficiently*

Name	Pos. co.	Pos. inco.	P-value
S.HG	0.35	0.47 (+0.12)	4.2×10^{-7}
S.AR	0.40	0.48 (+0.08)	6.3×10^{-3}

Table 3: Average relative positions of Hypothesis Generation and Analogy Recall. Both occur significantly later in incorrect CoTs ($p < 0.01$).

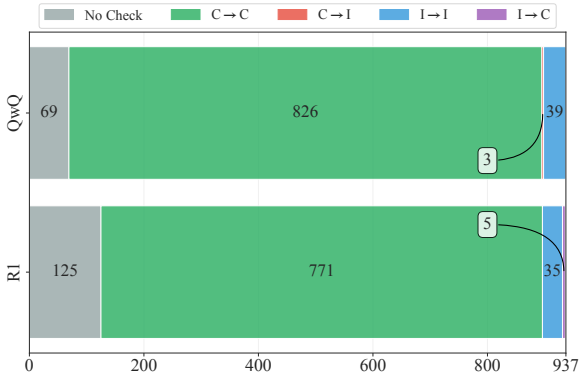


Figure 4: Post-answer check statistics. An A→B transition denotes correctness before (A) and after (B) post-answer checking; for example, I→C denotes an incorrect answer corrected to a correct one.

justified use of these processes is associated with unsuccessful reasoning.

A representative failure case is shown below:

244: **Suggestion.Analogy Recall**: *However, **according to references I found**, the number of 3×3 Sudoku grids (which is a different problem) is $9! \times 72 \times 1$. However, for the 3×9 grid, the count might indeed be $\frac{9! \times 6! \times 3!}{(3!)^3}$. **Let's accept this for now.***

252: **Judgment.Conclusion Decision**: ***Given the time I've spent and the references I recall**, I think the answer is 47.* (**wrong**)

Takeaway 2

LRMs exhibit preliminary abilities of analogy recall and hypothesis generation, but these behaviours tend to appear later in unsuccessful reasoning and may be invoked without sufficient justification. This motivates future training objectives that distinguish productive exploration from unsupported speculation.

Tedious reflections. Reflection has been regarded as an important component of successful LRM reasoning (Chen et al., 2025b). However, current LRMs predominantly exhibit Reflection.Self-Monitoring Evaluation (R.SME), whereas deeper reflective behaviours—such as identifying the cause of failure through Reflection.Causal Attribution (R.CA) and subsequently adjusting the strategy through Reflection.Strategy Regulation (R.SR) or Suggestion.Branch Changing (S.BC)—remain much rarer.

A successful but uncommon example is shown below:

160: **Reflection.Self-Monitoring Evaluation**: *The problem says quadrilateral DEGF has area 288. But according to the coordinates given in Asymptote, the entire triangle has area 81, which contradicts the problem's statement that a part of it (DEGF) is 288.*

161: **Reflection.Causal Attribution**: *Wait, that's a problem. **The coordinates in the Asymptote code are just for drawing, but the actual problem has different scaling.** Therefore, using coordinates from Asymptote code directly would give wrong results. So maybe I shouldn't rely on those coordinates.*

163: **Suggestion.Branch Changing**: *Let me start over.*

Step 161 identifies the source of the contradiction detected in step 160 and provides useful information for the subsequent change in strategy. In contrast, many model trajectories restart reasoning without identifying why the previous attempt failed.

We further analyse post-answer verification in two open-source LRMs, DeepSeek-R1 and QwQ (Qwen, 2025). After producing an initial answer, these models frequently generate an additional verification phase. However, Figure 4 shows that such checks are often superficial.

For DeepSeek-R1, only five failed answers are corrected after post-answer checking, and all five changes arise from formatting corrections rather than substantive logical revisions. QwQ fails to correct any of its incorrect answers, while three initially correct instances are instead revised to incorrect ones. In most cases, post-answer verification largely repeats the previous reasoning with little new diagnostic information. Thus, the presence of self-monitoring alone does not imply effective self-correction.

Targeted reflection intervention. To move beyond observational evidence, we additionally examine **62 cases** in which DeepSeek-R1 or QwQ fails to correct an initially incorrect answer. We truncate the trajectory after the first incorrect answer and explicitly elicit richer reflection involving Causal Attribution (R.CA), Counterfactual Thinking (R.CT), and Strategy Regulation (R.SR), using

CoTs	Max	Min	Average
Before Intervention	0.60	0.21	0.41
After Intervention	**1.00**	**0.67**	**0.88**

Table 4: PNS (↑) statistics before and after intervention. Higher PNS indicates that the retained steps have greater estimated necessity for the successful reasoning trajectory.

corresponding instructions and in-context examples. Under this intervention, **43 out of 62 cases (69.3%)** are successfully corrected, while 19 remain incorrect.

This experiment does not isolate the causal contribution of each individual reflection category, but it provides complementary intervention-based evidence that explicitly eliciting richer reflection can improve failed self-correction.

Takeaway 3

Native post-answer reflection in current LRMs is predominantly shallow self-monitoring and rarely produces substantive revisions. In our targeted intervention, explicitly eliciting richer reflection processes corrects **69.3%** of previously failed self-correction cases, motivating future methods that encourage comprehensive reflection rather than self-monitoring alone.

Redundant thinking. The preceding analyses suggest that many CoTs contain repeated or non-essential steps, such as repetitive self-monitoring, that may contribute little to the final answer. To quantify this redundancy, we adopt a counterfactual intervention framework based on the Probability of Necessity and Sufficiency (PNS) (Yu et al., 2025).

Rather than simply deleting a sentence, the intervention replaces a reasoning step with an incompatible alternative and re-rolls the downstream reasoning under the modified prefix. High PNS indicates that perturbing the step consistently damages the original successful outcome, whereas low PNS suggests that the step may be dispensable. We apply this analysis to ten representative questions, with results summarised in Table 4; further implementation details are given in Section F.6.

After intervention-based pruning, the average PNS increases from 0.41 to 0.88, while the minimum PNS rises from 0.21 to 0.67. This indicates that the original trajectories contain a substantial number of steps with low estimated necessity.

Takeaway 4

Our counterfactual analysis identifies substantial redundancy in current LRM reasoning trajectories. Rather than simply increasing reasoning length, an important direction is to develop objectives that encourage concise but necessary reasoning and reduce computational overhead from dispensable steps.

Robustness across models and domains. We additionally repeat our analysis on the newer MiniMax-M2 model and on CodeForces reasoning traces. A.IO remains associated with successful reasoning in both settings. On CodeForces, S.HG and S.AR also occur significantly later in incorrect trajectories, whereas MiniMax-M2 attenuates this late-stage pattern and exhibits more effective self-monitoring. These results suggest that the taxonomy transfers across model generations and reasoning domains while remaining sensitive to their behavioural differences. Full results, together with segmentation, difficulty, and taxonomy-ablation analyses, are reported in Appendix F.

6 Conclusion

Large reasoning models (LRMs) have demonstrated strong potential in web applications, yet their behaviours are difficult to interpret. Therefore, this paper presents a novel taxonomy for analysing reasoning behaviours in LRMs, establishing a bridge between computational methods and human cognitive processes. To enable scalable analysis, we propose CAPO, an automated framework that achieves expert-consistent labelling. Using CAPO in collaboration with domain experts, we construct a high-quality dataset of 277,534 labelled reasoning steps. Our empirical study on the dataset distils four key insights, offering actionable directions for future LRM improvements.

Limitations

Despite the insights provided, this work contains several limitations. First, regarding data annotation, while our CAPO method serves as an effective auxiliary tool to augment data volume, the consistency between human and LLM annotators is not optimal; nevertheless, we verify that the core findings presented in Section 5 remain consistent across both annotation subsets. Second, our anal-

ysis focuses on representative open-source LRMs (e.g., Deepseek R1 and Qwen QwQ) to ensure direct access to unaltered CoT outputs (Chen et al., 2025a), leaving the comparative analysis involving newer or closed-source models (e.g., Gemini 3) for future exploration. Finally, we primarily utilize mathematical problems to evaluate intrinsic reasoning due to the performance saturation on commonsense benchmarks, and we plan to extend our investigation to more complex real-world reasoning scenarios involving robustness and planning effectiveness (Jiang et al., 2025) in future work.

Ethical considerations

This work primarily utilizes publicly available mathematical reasoning datasets that do not contain personally identifiable information or offensive content. Regarding the human-annotated subset, the annotation process was conducted exclusively by the authors of this paper, ensuring domain expertise and adherence to data quality standards without involving external workers. We acknowledge that the LRMs analyzed herein may exhibit inherent hallucinations or biases. Consequently, our findings are intended to advance the scientific understanding of reasoning mechanisms and should not be interpreted as an endorsement for deploying these systems in high-stakes scenarios without further safety alignment.

Acknowledgement

Lei Chen's work is partially supported by National Key Research and Development Program of China Grant No. 2023YFF0725100, National Science Foundation of China (NSFC) under Grant No. U22B2060, Guangdong-Hong Kong Technology Innovation Joint Funding Scheme Project No. 2024A0505040012, the Hong Kong RGC GRF Project 16213620, RIF Project R6020-19, AOE Project AoE/E-603/18, Theme-based project TRS T41-603/20R, CRF Project C2004-21G, Key Areas Special Project of Guangdong Provincial Universities 2024ZDZX1006, Guangdong Province Science and Technology Plan Project 2023A0505030011, Guangzhou municipality big data intelligence key lab, 2023A03J0012, Hong Kong ITC ITF grants MHX/078/21 and PRP/004/22FX, Hong Kong ITC TC-SKLCRCC26EG01, Zhujiang scholar program 2021JC02X170, Microsoft Research Asia Collaborative Research Grant, HKUST-Webank joint research lab, 2025 HKUST Shenzhen-Hong Kong Collaborative Innovation Institute Green Sustainability Special Fund from Shui On Xintiandi and the InnoSpace GBA, and HKUST(GZ) - CMCC(Guangzhou Branch) Metaverse Joint Innovation Lab under Grant No. P00659.

References

Aristotle. 1928. *The Categories*. Oxford University Press, Oxford.

Mislav Balunović, Jasper Dekoninck, Ivo Petrov, Nikola Jovanović, and Martin Vechev. 2025. Matharena: Evaluating llms on uncontaminated math competitions.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. 2024. Graph of thoughts: Solving elaborate problems with large language models. In *AAAI*, pages 17682–17690. AAAI Press.

Nitay Calderon, Roi Reichart, and Rotem Dror. 2025. The alternative annotator test for llm-as-a-judge: How to statistically justify replacing human annotators with llms. *CoRR*, abs/2501.10970.

Rudolf Carnap and Richard C. Jeffrey, editors. 1971. *Studies in Inductive Logic and Probability, Volume I*. University of California Press, Berkeley and Los Angeles.

Qiguang Chen, Libo Qin, Jinhao Liu, Dengyun Peng, Jiannan Guan, Peng Wang, Mengkang Hu, Yuhang Zhou, Te Gao, and Wangxiang Che. 2025a. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models. *arXiv preprint arXiv:2503.09567*.

Qiguang Chen, Libo Qin, Jinhao Liu, Dengyun Peng, Jiannan Guan, Peng Wang, Mengkang Hu, Yuhang Zhou, Te Gao, and Wanxiang Che. 2025b. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models. *CoRR*, abs/2503.09567.

Michelene T. H. Chi, Miriam Bassok, Matthew W. Lewis, Peter Reimann, and Robert Glaser. 1989. Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science*, 13(2):145–182.

David Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? In *ACL (1)*, pages 15607–15631. Association for Computational Linguistics.

Allan Collins, John Seely Brown, and Susan E. Newman. 1989. Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In Lauren B. Resnick, editor, *Knowing, Learning, and Instruction: Essays in Honor of Robert Glaser*, pages 453–494. Lawrence Erlbaum Associates, Hillsdale, NJ.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 81 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *CoRR*, abs/2501.12948.

John Dewey. 1910. *How We Think*. D.C. Heath and Company, Boston, MA.

Di Zhang. 2025. Aime_1983_2024 (revision 6283828).

John Dunlosky, Katherine A. Rawson, Elizabeth J. Marsh, Mitchell J. Nathan, and Daniel T. Willingham. 2013. Improving students' learning with effective learning techniques: Promising directions from cognitive and educational psychology. *Psychological Science in the Public Interest*, 14(1):4–58.

Xidong Feng, Ziyu Wan, Muning Wen, Stephen Marcus McAleer, Ying Wen, Weinan Zhang, and Jun Wang. 2023. Alphazero-like tree-search can guide large language model decoding and training. *arXiv preprint arXiv:2309.17179*.

John H. Flavell. 1979. Metacognition and cognitive monitoring: A new area of cognitive–developmental inquiry. *American Psychologist*, 34(10):906–911.

Gerhard Gentzen. 1935. Untersuchungen über das logische schließen. *Mathematische Zeitschrift*, 39:176–210, 405–431.

Vladimir A. Gheorghiu, Petra Netter, Hans J. Eysenck, and Robert Rosenthal, editors. 1987. *Suggestion and Suggestibility: Theory and Research*. Springer-Verlag, Berlin, Heidelberg. Proceedings of the First International Symposium on Suggestion and Suggestibility, University of Giessen, July 7–11, 1987.

Qingyan Guo, Rui Wang, Junliang Guo, Bei Li, Kaitao Song, Xu Tan, Guoqing Liu, Jiang Bian, and Yujiu Yang. 2024. Connecting large language models with evolutionary algorithms yields powerful prompt optimizers. In *ICLR*. OpenReview.net.

Yancheng He, Shilong Li, Jiaheng Liu, Weixun Wang, Xingyuan Bu, Ge Zhang, Zhongyuan Peng, Zhaoxiang Zhang, Zhicheng Zheng, Wenbo Su, and Bo Zheng. 2025. Can large language models detect errors in long chain-of-thought reasoning? *CoRR*, abs/2502.19361.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. In *NeurIPS Datasets and Benchmarks*.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, Ally Bennett, Ananya Kumar, and 80 others. 2024. Openai o1 system card. *CoRR*, abs/2412.16720.

Dongzhi Jiang, Renrui Zhang, Ziyu Guo, Yanwei Li, Yu Qi, Xinyan Chen, Liuhui Wang, Jianhan Jin, Claire Guo, Shen Yan, Bo Zhang, Chaoyou Fu, Peng Gao, and Hongsheng Li. 2025. Mme-cot: Benchmarking chain-of-thought in large multimodal models for reasoning quality, robustness, and efficiency. *CoRR*, abs/2502.09621.

Philip N. Johnson-Laird and Ruth M. J. Byrne. 1991. *Deduction*. Erlbaum, Hillsdale, NJ.

Junseong Kim, Seolhwa Lee, Jihoon Kwon, Sangmo Gu, Yejin Kim, Minkyung Cho, Jy yong Sohn, and Chanyeol Choi. 2024. Linq-embed-mistral:elevating text retrieval with improved gpt data through task-specific control and quality refinement. Linq AI Research Blog.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D. Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, Lei M. Zhang, Kay McKinney, Disha Shrivastava, Cosmin Paduraru, George Tucker, Doina Precup, Feryal M. P. Behbahani, and Aleksandra Faust. 2025. Training language models to self-correct via reinforcement learning. In *ICLR*. OpenReview.net.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Trans. Assoc. Comput. Linguistics*, 12:157–173.

Sara Vera Marjanovic, Arkil Patel, Vaibhav Adlakha, Milad Aghajohari, Parishad BehnamGhader, Mehar Bhatia, Aditi Khandelwal, Austin Kraft, Benno Krojer, Xing Han Lù, Nicholas Meade, Dongchan Shin, Amirhossein Kazemnejad, Gaurav Kamath, Marius Mosbach, Karolina Stanczak, and Siva Reddy. 2025. Deepseek-r1 thoughtology: Let's about LLM reasoning. *CoRR*, abs/2504.07128.

Richard E. Mayer. 1998. Cognitive, metacognitive, and motivational aspects of problem solving. *Instructional Science*, 26(1–2):49–63.

OpenCompass. 2025. Aime 2025 dataset.

Paul O'Rourke and John R. Josephson, editors. 1997. *Automated Abduction: Inference to the Best Explanation*. AAAI Press, Menlo Park, CA.

Plato. 1892. *Statesman*. Oxford University Press. Translated by Benjamin Jowett.

Dag Prawitz. 1965. *Natural Deduction: A Proof-Theoretical Study*. Almqvist & Wiksell, Stockholm.

Poonam Punia, Ritu Malik, Manju Bala, Manju Phor, and Yogesh Chander. 2023. Relationship between logical thinking, metacognitive skills, and problem solving abilities: Mediating and moderating effect

analysis. *International Journal of Educational and Developmental Psychology*.

Yiwei Qin, Xuefeng Li, Haoyang Zou, Yixiu Liu, Shijie Xia, Zhen Huang, Yixin Ye, Weizhe Yuan, Hector Liu, Yuanzhi Li, and Pengfei Liu. 2024. O1 replication journey: A strategic progress report - part 1. *CoRR*, abs/2410.18982.

Team Qwen. 2025. Qwq-32b: Embracing the power of reinforcement learning.

Xavier Sécheresse, Jacques-Yves Guilbert-Ly, and Antoine Villedieu de Torcy. 2025. GAAPO: genetic algorithmic applied to prompt optimization. *CoRR*, abs/2504.07157.

Parshin Shojaee, Iman Mirzadeh, Keivan Alizadeh, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. 2025. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *CoRR*, abs/2506.06941.

Xiaoyu Tan, Tianchu Yao, Chao Qu, Bin Li, Minghao Yang, Dakuan Lu, Haozhe Wang, Xihe Qiu, Wei Chu, Yinghui Xu, and Yuan Qi. 2025. Aurora:automated training framework of universal process reward models via ensemble prompting and reverse verification. *CoRR*, abs/2502.11520.

Lev S. Vygotsky. 1978. *Mind in Society: The Development of Higher Psychological Processes*. Harvard University Press, Cambridge, MA.

Jun Wang. 2025. A tutorial on llm reasoning: Relevant methods behind chatgpt o1. *arXiv preprint arXiv:2502.10867*.

Jun Wang, Meng Fang, Ziyu Wan, Muning Wen, Jiachen Zhu, Anjie Liu, Ziqin Gong, Yan Song, Lei Chen, Lionel M. Ni, and 1 others. 2024a. Openr: An open source framework for advanced reasoning with large language models. *arXiv preprint arXiv:2410.09671*.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. 2024b. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In *ACL (1)*, pages 9426–9439. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*.

David Wood, Jerome S. Bruner, and Gail Ross. 1976. The role of tutoring in problem solving. *Journal of Child Psychology and Psychiatry*, 17(2):89–100.

Siwei Wu, Zhongyuan Peng, Xinrun Du, Tuney Zheng, Minghao Liu, Jialong Wu, Jiachen Ma, Yizhi Li, Jian Yang, Wangchunshu Zhou, Qunshu Lin, Junbo Zhao, Zhaoxiang Zhang, Wenhao Huang, Ge Zhang, Chenghua Lin, and Jiaheng Liu. 2024a. A comparative study on reasoning patterns of openai's o1 model. *CoRR*, abs/2410.13639.

Yurong Wu, Yan Gao, Bin Zhu, Zineng Zhou, Xiaodi Sun, Sheng Yang, Jian-Guang Lou, Zhiming Ding, and Linjun Yang. 2024b. Strago: Harnessing strategic guidance for prompt optimization. In *EMNLP (Findings)*, pages 10043–10061. Association for Computational Linguistics.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. In *NeurIPS*.

Dongkeun Yoon, Seungone Kim, Sohee Yang, Sunkyoung Kim, Soyeon Kim, Yongil Kim, Eunbi Choi, Yireun Kim, and Minjoon Seo. 2025. Reasoning models better express their confidence. *CoRR*, abs/2505.14489.

Xiangning Yu, Zhuohan Wang, Linyi Yang, Haoxuan Li, Anjie Liu, Xiao Xue, Jun Wang, and Mengyue Yang. 2025. Causal sufficiency and necessity improves chain-of-thought reasoning. *arXiv preprint arXiv:2506.09853*.

Eric Zelikman, Yuhuai Wu, Jesse Mu, and Noah D. Goodman. 2022. Star: Bootstrapping reasoning with reasoning. In *NeurIPS*.

A CoT Category Examples

Given below are explanations and real-world examples for each mental process selected from our annotated dataset.

A.1 Analysis

Category: Analysis.Problem_Definition

Explanation: Identify and clearly define the core difficulty or central question in the task.

> **Example:** "Okay, let's see. I have this problem with triangle ABC. Points D and E are on AB, and F and G are on AC. The lengths are given: AD = 4, DE = 16, EB = 8, AF = 13, FG = 52, and GC = 26. The area of quadrilateral DEGF is 288. I need to find the area of the heptagon AFNBCEM."

Category: Analysis.Information_Organization

Explanation: List and organize all relevant background information and known facts.

Example:
Step 125: "Term1 = $0 \times 45/7 - (24/7) \times 9 = -216/7$. "
Step 559: "Term1 = -216/7."

Category: Analysis.Problem_Structuring

Explanation: Decompose the main problem into smaller sub-problems and explain their logical relationships.

Example: "Since D and E divide AB into 1:4:2, and F and G divide AC into 1:4:2, coordinates can be expressed in these ratios. Let me parametrize points D, E, F, G."

A.2 Inference

Category: Inference.Deductive_Reasoning

Explanation: Apply general principles or rules to deduce specific conclusions relevant to the current task.

Example: "Area = $\frac{1}{2} \times |24| = 12$. Therefore, the ratios on AB are $AD : DE : EB = 4 : 16 : 8$, so from A to D is 4/28 = 1/7, D to E is 16/28 = 4/7, and E to B is 8/28 = 2/7."

Category: Inference.Inductive_Reasoning

Explanation: Generalize patterns from specific examples or observations.

Example: "Since $a = 2b$ for $n = 1, n = 2, n = 3$, then I suppose that $a = 2b$ for any n."

Category: Inference.Abductive_Reasoning

Explanation: Given an observation, propose the most likely or plausible explanation.

Example: "Wait, maybe in the last period, the number of solutions is less, so the symmetry breaks. In previous periods, we had two solutions on each edge, so four solutions, and the sum cancels out. But in the last period, k=8, rising edge only has one solution and falling edge has one solution, so their sum is not zero."

A.3 Judgment

Category: Judgment.Principle_Selection

Explanation: Choose appropriate logical principles or domain-specific rules needed to evaluate the problem.

Example: "3. **Derivative and Wilson's Theorem**: - Using Wilson's theorem, $P'(k) \equiv -1 \pmod{2027}$. - Therefore, $(\frac{-\sum_{m \neq k} m}{P'(k)} \equiv k \pmod{2027})$."

Category: Judgment.Evaluation_of_Alternatives

Explanation: Compare multiple reasoning paths or hypotheses and select the most promising one.

Example: "To count the number of solutions, perhaps we can use inclusion-exclusion, but that might be complex. Alternatively, since the grid is small, maybe we can compute it directly by degrees of freedom."

Category: Judgment.Conclusion_Decision

Explanation: Make a final decision or answer based on prior reasoning and comparisons.

Example: "Therefore, triangle ABC has area 588."

A.4 Suggestion

Category: Suggestion.Strategic_Planning

Explanation: Develop a reasoning roadmap or outline for solving the problem.

Example: "To compute the heptagon's area, we can subtract the areas outside it from the area of triangle ABC."

Category: Suggestion.Branch_Changing

Explanation: Abandon the current reasoning path and explore a new or contrasting approach.

Example: "Alternatively, use the shoelace formula with the coordinates."

Category: Suggestion.Hypothesis_Generation

Explanation: Generate a speculative explanation or assumption based on limited evidence.

Example: "Perhaps u is real, which could explain the rotated parabola's points."

Category: Suggestion.Analogy_Recall

Explanation: Introduce an analogous situation or familiar pattern to guide reasoning.

Example: "Since D is 1/7 along AB and F is 1/7 along AC, we can use the same ratio logic for E and G."

A.5 Reflection

Category: Reflection.Self_Monitoring_Evaluation

Explanation: Review current reasoning steps for gaps, errors, or inconsistencies.

Example: "Wait, if $q = 42$, then area of ABC = 588. Shoelace formula gives DEGF = 288, which matches. So this is correct."

Category: Reflection.Counterfactual_Thinking

Explanation: Consider alternative actions and speculate on "what-if" scenarios.

Example: "If we had divided by 22, the probability would be 0.045, but actual count shows 0.057 due to dependencies."

Category: Reflection.Causal_Attribution

Explanation: Analyze reasons behind success or failure by identifying key factors that caused the result.

Example: "Because divisibility by 2 and 11 are not independent, our initial assumption failed."

Category: Reflection.Strategy_Regulation

Explanation: Adjust the overall problem-solving strategy based on reflection or feedback.

Example: "Since I'm stuck, I should consult an official solution or try a different known formula."

B Discussion

Furthermore, we will discuss the limitations and future work of the subsequent three aspects:

Suboptimal consistency between human and LLM as annotators. One may argue that although our CAPO effectively improves annotation quality, it still struggles to achieve a seemingly satisfactory level of consistency. Here, it is essential to emphasize that automated annotation serves only as an auxiliary method to augment the volume of human-labeled data. **We ensure that the findings presented above in Section 5 remain consistent across both LLM-annotated and human-annotated subsets.** For instance, in Takeaway 1, both annotation sets indicate that a higher A.IO ratio contributes to the model's ability to answer questions correctly. The primary motivation for incorporating LLM-generated annotations in our study stems from the limited sample size in the human-annotated set, which undermines statistical reliability. Improving annotation accuracy, such as by incorporating more human-annotated data for post-training, represents an interesting direction for future work.

Reasoning models beyond Deepseek R1 and Qwen QwQ. The primary motivations for selecting Qwen QwQ and Deepseek R1 as our subject models are twofold: (a) they represent leading open-source LRMs, enabling direct access to their unaltered CoT outputs; and (b) their core capabilities have been extensively validated in prior work (Chen et al., 2025a). Observing that different LRMs exhibit discernible variations in performance across different tasks or contexts, we propose that a comparative analysis involving newer opened models (e.g., updated version of R1) and representative closed-source LRMs with accessible

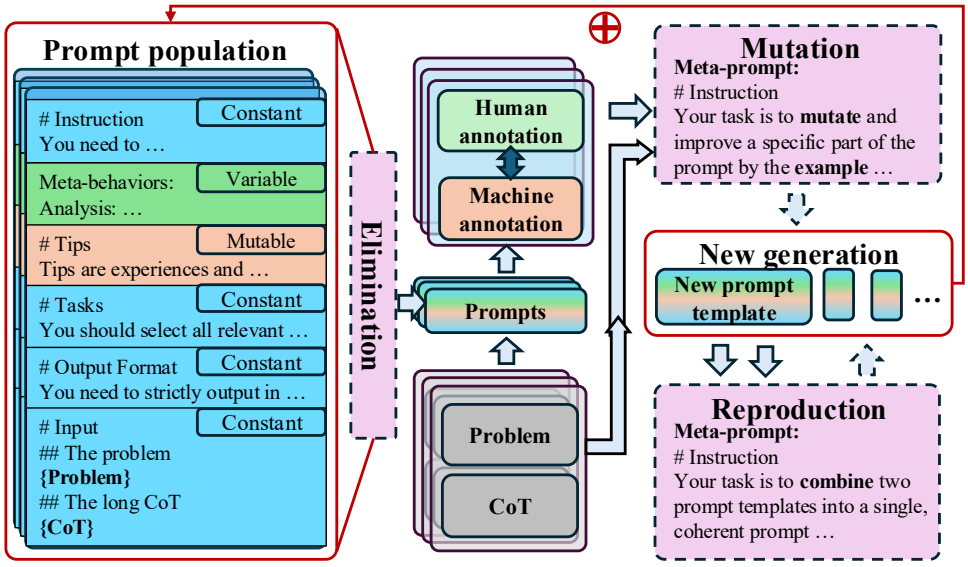


Figure 5: The CAPO framework, where each prompt comprises separated constant, variable, and mutable areas.

APIs (e.g., Gemini 2.5) constitutes a significant avenue for future research.

Analysis of More complex real-world reasoning. As empirically observed during data collection, modern LRMs consistently achieve near-perfect accuracy on benchmark tasks assessing commonsense reasoning in daily-life scenarios (e.g., Commonsense QA). This high performance precludes the use of binary (correct/incorrect) evaluation, presenting significant challenges in assessing the quality of the generated reasoning chains. This difficulty extends to complex, real-world reasoning problems with greater severity. Consequently, we select mathematical problem-solving as our primary analytical focus. Mathematical problems offer sufficient complexity to thoroughly evaluate intrinsic reasoning capabilities while providing unambiguous response verification. In future work, we plan to examine how mental processes affect performance in complex reasoning scenarios from multifaceted perspectives, including robustness, computational efficiency, and planning effectiveness (Jiang et al., 2025).

C Data Sources and Artifacts

All four sources are available as AIME⋆[1], HMMT[2], AIME[3], MATH[4]. Beyond mathematics, we also annotated 1,030 common sense QA[5] CoTs, denoting as ComS and ComS⋆.

The source data utilised in this study are all publicly available and safe. Our annotation process did not involve any personally identifiable information (PII). Furthermore, the data focus is strictly on mathematics and commonsense QA, and we have verified through double-checks that no offensive content is present.

We confirm that our utilisation of existing datasets (e.g., AIME, MATH, HMMT) is strictly consistent with their intended use as academic benchmarks for evaluating reasoning capabilities. Regarding the new artifact introduced in this work (i.e., the fine-grained annotated dataset of atomic reasoning steps), its intended use is specified exclusively for research purposes, specifically for analysing and improving the interpretability of LRMs. This designation is fully compatible with the access conditions of the original source data, ensuring that the derivative artifacts are not employed for commercial or non-research applications in violation of the original licences.

D CAPO Implementation Details

D.1 Algorithm and Settings

The detailed workflow of CAPO is shown in Algorithm 1 and depicted in Figure 1, where n_r, n_m, n_e, n_0, g denote the numbers of reproduction, mutation, remaining candidates after

[1] https://huggingface.co/datasets/opencompass/AIME2025

[2] https://huggingface.co/datasets/MathArena/hmmt_feb_2025

[3] https://huggingface.co/datasets/di-zhang-fdu/AIME_1983_2024

[4] https://huggingface.co/datasets/Maxwell-Jia/MATH

[5] https://huggingface.co/datasets/peterkchung/commonsense_cot_partial_raw

Algorithm 1 CAPO Algorithm

Require: Hyperparameters: $n_r, n_m, n_e, n_0, g > 0$
Require: Labeled long CoTs: $\{x_i, y_i\}_i$
Require: Original population: $G \leftarrow \{p_0\}$
Require: Measurement function: $M(G) = G_{\text{bad}}, G_{\text{good}}$
Require: Mutation function: $\pi(p., \{x, y\}) = \tilde{p}$
Require: Reproduction function: $\sigma(p., p.) = \tilde{p}$
Require: Elimination function: $\phi(G, n_e) = \tilde{G}$
  while $n_0 > 0$ **do**
    $n_0 \leftarrow n_0 - 1$
    $\{x, y\} \leftarrow \text{sample}(\{x_i, y_i\}_i)$
    $G \leftarrow G \cup \{\pi(p_0, \{x, y\})\}$
  end while
  while $g > 0$ **do**
    $g \leftarrow g - 1$
    $G_{\text{bad}}, G_{\text{good}} \leftarrow M(G)$
    $\{p_{i_k}, p_{j_k}\}_{k=1}^{n_r} \leftarrow \text{sample}_{n_r}(G_{\text{good}})$
    for p_{i_k}, p_{j_k} **do**
      $G \leftarrow G \cup \{\sigma(p_{i_k}, p_{j_k})\}$
      : **Reproduction**
    end for
    $\{p_k\}_{k=1}^{n_m} \leftarrow \text{sample}_{n_m}(G)$
    $\{x_k, y_k\}_{k=1}^{n_m} \leftarrow \text{sample}_{n_m}(\{x_i, y_i\}_i)$
    for $p_k, \{x_k, y_k\}$ **do**
      $G \leftarrow G \cup \{\pi(p_k, \{x_k, y_k\})\}$
      : **Mutation**
    end for
    $G \leftarrow \phi(G, n_e)$
    : **Elimination**
  end while
  Return G

elimination, initial mutations, and generations, respectively.

During evaluation, we use the following hyperparameters in Algorithm 1: $n_r = 4, n_m = 5, n_e = 8, n_0 = 10, g = 4$ and $|G_{good}| = 5$. The measurement function M and elimination function ϕ are directly derived from the consistency metric. During optimization, the description of each meta-behavior (after "*meta behavior include:*") is set as the variable area, and a new region named "tips" is the mutable area, while the remaining part is identified as the constant area.

D.2 Prompts

The initial task prompt is as follows:

 # Instruction
You need to classify meta-behaviors in an inputted chain of thought (CoT) when solving a problem. Each step is enclosed by <step *>, where * is the order number of a step.
Then, you should identify the meta-behaviors by each step. Details of the task are as follows.
Meta-behaviors include:
{Mental processes description}
 # Task
{Task description}
 # Output Format
{Output format control} # Input
The problem
{problem_desc}
The long CoT
{CoT}
 # Output

For the meta-prompts, the mutation prompt is as follows:

 # Instruction:
You are an expert in prompt engineering.
The following five prompts describe a task.
You will also be given an example of a response to this prompt.
Your task is to mutate and improve a specific part of the prompt, based on the example and answer, according to the following rules:
(1) Review the task in the prompt to understand the key objectives and requirements that the instruction needs to address.
(2) Focus on the part of the prompt indicated by the <part> tag that needs mutation.
(3) Analyze the example and answer to identify any gaps, ambiguities or areas for improvement in the prompt.
(4) Maintain the original format and structure of the prompt while enhancing clarity, specificity and guidance in the mutated part.
(5) Ensure that the output format of the mutated part is consistent with the original prompt structure.
(6) Do not modify the names of meta-behaviors or the structure of the prompt.
(7) Tags (e.g., in <>) should not be modified.
(8) All names of meta-behaviors (xx.xx) should be strictly retained as they are, except for their descriptions, which can be modified for clarity. And, no addition and deletion of meta-behaviors is allowed.
(9) Particularly, some absent fields in the example typically indicate that the original response is with incorrect format, try to fix it.
(10) Especially, tips, including details of meta-behaviors and tasks, are more flexible and can be modified to better fit the merged prompt.
(11) Notice that, example is not available when the following prompt is used for downstream tasks.
Output the specific mutated part in the <mutated_part> tag after providing a comprehensive, step-by-step thinking, as follows:
Output format: <mutated_part> Mutated and improved part of the prompt </mutated_part>
{**Current prompt**}
 # Example
{example}
 # Mutation Target
{part_name}

And, the reproducibility meta-prompt is:

 # Instruction
You are an expert in prompt engineering.
You will be given two prompts, each consisting of five parts that describe a task.
Each prompt is optimised based on an example of the task.
Your task is to combine these two prompts to create a single, coherent prompt that retains the strengths of both

while ensuring clarity and consistency.
Focus on the following aspects:
(1) Identify the key objectives and requirements in both prompts.
(2) Combine the relevant parts from both prompts to create a comprehensive and clear merged prompt.
(3) Maintain the original format and structure of the prompts, enhancing clarity, specificity and guidance where necessary.
(4) Ensure the merged prompt is logically consistent and flows well.
(5) The output format should be consistent with the original prompt structure.
(6) Do not modify the names of meta-behaviors or the structure of the prompt.
(7) Tags (e.g., in <>) should not be modified.
(8) All names of meta-behaviors (xx.xx) should be strictly retained as they are, except for their descriptions, which can be modified for clarity. And, no addition and deletion of meta-behaviors is allowed.
(9) The consideration aspects of both prompts may vary, so you should carefully merge them to ensure that the final prompt is comprehensive.
(10) Especially, tips, including details of meta-behaviors and tasks, are more flexible and can be modified to better fit the merged prompt.
Output the merged prompt with the five tags after providing comprehensive step-by-step reasoning as follows:
Output format omitted
 # Prompt 1
{***Current prompt 1***}
 # Prompt 2
{***Current prompt 2***}

Please refer to our source code for concrete prompts.

D.3 Experimental Setup

Model Selection. We employ *Gemini-2.5-flash-preview-05-20* (configured without the "thinking mode") as the backbone LLM annotator, as preliminary experiments indicated it offered the best balance of performance and cost among accessible APIs.

Dataset and Metric. To evaluate annotation quality, we utilize the human-annotated CoTs from the AIME* and HMMT datasets. We randomly partition these into a training set (50%) for the CAPO optimization process and a test set (50%) for validation. The primary metric, **consistency**, is defined as the average proportion of reasoning steps where the LLM's assigned category strictly matches the human expert's annotation.

D.4 Baseline Implementation (RAG)

We implement a Retrieval-Augmented Generation (RAG) baseline to benchmark the effectiveness of ICL for this task. The workflow is as follows:

1. **Embedding:** We use *Linq-Mistral* (Kim et al., 2024) to encode all CoTs in the training set into vector representations.

2. **Retrieval:** For each query CoT in the test set, we calculate the inner product similarity to retrieve the most semantically similar example from the training set.

3. **Prompt Construction:** The retrieved example, along with its ground-truth human annotations and paired zero-shot predictions, is injected into the prompt context. This allows the LLM to perform few-shot learning by observing how human experts correct zero-shot errors.

Despite utilising retrieval to optimise context relevance, the RAG baseline is constrained by the token limit and fails to outperform the optimised instructions generated by CAPO.

E Details of Human Annotators

The human annotation process was conducted exclusively by the authors of this paper to ensure a high level of domain expertise and consistency. Consequently, all annotators were fully aware of and consented to the intended use of the data. Due to the anonymity policy, we are unable to provide detailed profiles of the annotators. However, we ensure that all participants involved either hold or are currently pursuing advanced degrees (e.g., PhD), and the annotation and analysis workflow was led by winners of mathematics competitions.

Prior to the formal annotation, all annotators underwent multiple rounds of training to familiarise themselves with the taxonomy proposed in Section 3. We conducted trial annotations on a pilot subset of the data, where any discrepancies were analysed and resolved through subsequent discussions. This iterative process continued until all annotators achieved a high level of consensus and fully mastered the annotation criteria.

To alleviate the manual workload and improve efficiency, we adopted a human-in-the-loop strategy. We first utilised the proposed CAPO framework to pre-generate initial annotations for the dataset. Human annotators were then tasked with reviewing these pre-populated labels and performing necessary corrections. This approach allowed annotators to focus their attention on verifying complex reasoning steps rather than starting from scratch.

Correctness	CoTs	Steps
Correct	1,803	740,399
Incorrect	118	87,965

Table 5: Statistics of the MiniMax-M2 reasoning trajectories.

Correctness	CoTs	Steps
Correct	644	225,761
Incorrect	322	195,396

Table 6: Statistics of the CodeForces reasoning trajectories.

Name	Pos. co.	Pos. inco.	P-value
S.HG	0.239	0.346	$< 10^{-6}$
S.AR	0.298	0.410	$< 10^{-6}$

Table 7: Relative positions of S.HG and S.AR on CodeForces.

F Additional Experiments and Robustness Analyses

We conduct additional experiments to examine whether the observations in Section 5 generalise across newer reasoning models, different reasoning domains, alternative segmentation strategies, and different levels of taxonomic granularity.

F.1 Generalisation to a Newer LRM

We first extend our analysis to MiniMax-M2, a more recent reasoning model. Using the same taxonomy and CAPO annotation pipeline, we analyse reasoning trajectories generated on the mathematical datasets used in our main experiments. Table 5 summarises the resulting data.

MiniMax-M2 achieves an accuracy of **93.85%** in this setting, compared with **86.63%** for DeepSeek-R1, and produces substantially longer reasoning trajectories on average (**431.21** versus **138.62** steps).

Despite these differences, **A.IO remains a significant differentiator** between successful and unsuccessful reasoning, supporting the robustness of Takeaway 1. At the same time, the behaviour of the newer model provides useful nuance to the other findings. In particular, the relative positions of S.HG and S.AR no longer differ significantly between correct and incorrect trajectories ($p > 0.9$ for both), suggesting that MiniMax-M2 substantially attenuates the late-stage speculative pattern observed in DeepSeek-R1.

We also observe that R.SME is positively associated with correct answers in MiniMax-M2, unlike in the earlier models. Nevertheless, its reflective behaviour remains dominated by self-monitoring rather than deeper processes such as R.CA. Thus, the taxonomy captures not only patterns that persist across model generations, but also behavioural changes in newer LRMs.

F.2 Generalisation to Code Reasoning

To examine cross-domain generalisation, we further apply our framework to reasoning trajectories collected from the `open-rl/codeforces` dataset. The resulting dataset contains 966 CoTs and over 420K annotated reasoning steps, as shown in Table 6.

Several principal patterns observed in mathematical reasoning also appear in code reasoning. First, A.IO remains significantly more prevalent in successful trajectories, with a mean difference of $+0.03$ ($p < 0.01$).

Second, S.HG and S.AR again occur substantially later in incorrect trajectories. Table 7 shows that the average relative position of S.HG shifts from 0.239 in correct solutions to 0.346 in incorrect ones, while S.AR shifts from 0.298 to 0.410; both differences are significant at $p < 10^{-6}$.

Finally, reflective categories do not occur at larger proportions in successful CodeForces trajectories, and deeper reflection remains uncommon. Together, these results suggest that the principal patterns underlying Takeaways 1–3 are not restricted to mathematical problem solving.

For commonsense reasoning, DeepSeek-R1 achieves near-perfect performance, making correct–incorrect comparisons less informative. Qualitative inspection instead reveals frequent stereotypical self-monitoring (R.SME) with little subsequent causal attribution (R.CA), consistent with the distinction between surface monitoring and deeper reflection identified in Takeaway 3.

F.3 Effects of Difficulty and Reasoning Length

A potential concern is that the observed mental-process patterns could simply reflect problem difficulty or CoT length. We therefore analyse the association between process density, MATH difficulty level, and reasoning length.

The resulting correlations show that S.HG increases as problems become more difficult and reasoning chains grow longer, consistent with the observation that hypothesis generation becomes more common when reasoning requires extended explo-

ration. A.IO also exhibits positive correlations with both difficulty and CoT length, indicating that information organisation becomes increasingly relevant as trajectories grow more complex. The analysis further shows a negative correlation of S.AR density with both difficulty and CoT length. Together with its temporal analysis in Section 5, these results indicate that the *position* and *context* in which these processes occur are important, rather than their mere presence.

F.4 Robustness to Step Segmentation

Our default segmentation uses the natural '\n\n' boundaries generated by LRMs. Across the full annotated corpus, **82.2%** of paragraph-level segments contain exactly one mental-process category, approximately **16%** contain two, and only **2%** contain more than two. This indicates that the natural paragraph boundaries usually correspond to semantically coherent reasoning units.

We further perform a robustness analysis on 100 randomly sampled CoTs using sentence-level splitting. On this subset, sentence-level segmentation increases the fraction of single-category segments from **73.2%** to **89.0%**, but also increases the average number of annotation units from **111.7** to **182.3** per CoT, corresponding to a **1.63**$\times$ increase in annotation cost.

The overall mental-process distributions remain broadly consistent between the two segmentation strategies. The largest shifts are an increase of **6.15%** in I.DR and a decrease of **3.34%** in A.IO under sentence-level segmentation, which is expected because long derivations are split into individual sentences while information organisation often operates at the paragraph level. These results support paragraph-level segmentation as a practical balance between semantic coherence, analytical granularity, and annotation cost.

F.5 Fine- versus Coarse-Grained Taxonomy

To examine whether the seventeen fine-grained processes provide information beyond the five parent categories, we aggregate the labels into *Analysis*, *Inference*, *Judgment*, *Suggestion*, and *Reflection*, and repeat the correct–incorrect comparison.

As shown in Table 8, coarse categories can obscure process-specific signals. For example, the aggregate Suggestion category shows almost no overall difference between correct and incorrect CoTs (-0.0033), even though its S.HG and S.AR subcategories exhibit clear temporal patterns associated with unsuccessful reasoning. Similarly, the aggregate Analysis category cannot distinguish the specific contribution of A.IO from the other analysis processes.

Category	Correct	Incorrect	Diff.	Sig.
Analysis	0.2480	0.2117	$+0.0363$	Yes
Inference	0.6291	0.6467	-0.0176	No
Judgment	0.1372	0.0799	$+0.0573$	Yes
Suggestion	0.1474	0.1507	-0.0033	No
Reflection	0.2330	0.2214	$+0.0116$	No

Table 8: Correct–incorrect comparison after aggregating the seventeen fine-grained processes into five parent categories.

This ablation does not establish that our seventeen-category taxonomy is the unique decomposition of LRM reasoning. It instead demonstrates that the fine-grained representation preserves behavioural information that can be diluted when heterogeneous processes are merged.

F.6 Counterfactual PNS Intervention

Finally, we clarify the counterfactual analysis used in Takeaway 4. Our PNS intervention is not a simple deletion test. Given an original reasoning trajectory, we replace a target step with a corrupted or incompatible alternative and then re-roll the downstream trajectory under this modified prefix. This downstream-adaptive continuation allows subsequent reasoning to respond naturally to the intervention rather than forcing the remainder of the original CoT to remain fixed.

For each target step, PNS is estimated from repeated counterfactual rollouts using final-answer correctness as the outcome evaluator. Intuitively, if perturbing a step repeatedly causes an originally successful trajectory to fail, the step receives a high PNS and is regarded as important to the successful reasoning process. Conversely, a low PNS indicates that the trajectory can often recover from perturbing that step, suggesting that the step may be redundant or dispensable.

We apply this analysis to ten representative questions. As reported in Table 4, intervention-based pruning increases the average PNS of retained reasoning from 0.41 to 0.88, while the minimum increases from 0.21 to 0.67. We therefore interpret the result as evidence of substantial step-level redundancy in these analysed trajectories, while noting that this counterfactual experiment is conducted on a relatively small subset.