Self-Awareness before Action: Mitigating Logical Inertia via Proactive Cognitive Awareness

Anonymous ACL submission

Abstract

Large language models perform well on many reasoning tasks, yet they often lack awareness of whether their current knowledge or reasoning state is complete. In non-interactive puzzle settings, the narrative is fixed and the underlying structure is hidden; once a model forms an early hypothesis under incomplete premises, it can propagate that error throughout the reasoning process, leading to unstable conclusions. To address this issue, we propose SABA, a reasoning framework that explicitly introduces self-awareness of missing premises before making the final decision. SABA formulates reasoning as a recursive process that alternates between structured state construction and obstacle resolution: it first applies Information Fusion to consolidate the narrative into a verifiable base state, and then uses Query-driven Structured Reasoning to identify and resolve missing or underspecified premises by turning them into queries and progressively completing the reasoning state through hypothesis construction and state refinement. Across multiple evaluation metrics, SABA achieves the best performance on all three difficulty splits of the non-interactive Detective Puzzle benchmark, and it also maintains leading results on multiple public benchmarks.[1]

1 Introduction

Large language models have shown strong ability in multi-step reasoning and narrative understanding. In interactive settings such as social games, agents can acquire new information through dialogue and revise their beliefs over time (Zhang et al., 2024; Song et al., 2025; Wang et al., 2023a; Zhu et al., 2025; Wu et al., 2024). In contrast, in non-interactive puzzle settings, the narrative is fixed and no new information can be obtained. The model must recover the hidden truth only from a long text that contains implicit clues, missing links, and distractors. In this regime, the main challenge is not how to ask for information, but how to locate, align, and connect the information that already exists in the narrative (Liu et al., 2023). A small early mistake can remain uncorrected and can guide all later reasoning, which often leads to unstable and incorrect conclusions (Ji et al., 2023; Dziri et al., 2023). Existing work on abductive and long-context reasoning reports that current models still struggle under this form of information asymmetry (Del and Fishel, 2023; Piękos et al., 2021; Wang et al., 2024). This observation motivates the need for a reasoning process that can correct such early errors.

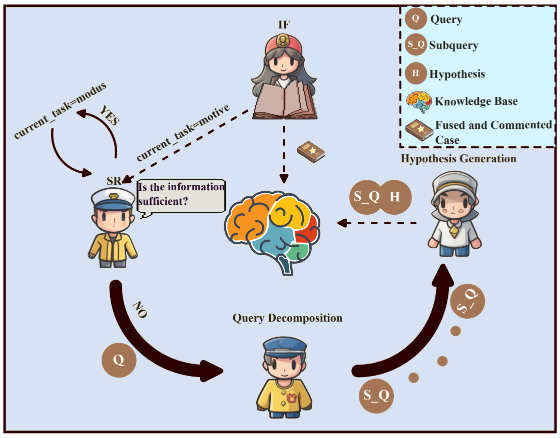


Figure 1: Overview of the SABA Framework.

Most existing reasoning paradigms are not well suited to this setting. Prompt-based methods such as chain-of-thought tend to commit to an early hypothesis and then expand it, even when the initial premise is weak (Wei et al., 2022; Turpin et al., 2023). Decomposition methods introduce intermediate steps, but they often lose global coherence when the narrative is long and the evidence is scattered (Zhou et al., 2023; Khot et al., 2023). Refinement-based methods revise an answer after it is produced, but they often justify the same early mistake instead of triggering a full re-evaluation,

---

[1] Code is available at `https://anonymous.4open.science/r/SABA-B6DA`

which leads to confirmation bias (Huang et al., 2024; Stechly et al., 2023). These limitations suggest that a reliable agent should not start from answering the task, but should first examine whether its current understanding is complete and consistent (Shinn et al., 2023). This perspective shifts the focus from direct prediction to state assessment.

Based on this view, we propose SABA, a reasoning framework that treats the model as a system that audits its own knowledge state before making a decision. As shown in Figure 1, the core idea of SABA is to perform long-horizon reasoning through a recursive control loop that alternates between structured state construction and obstacle-driven reasoning. SABA consists of two modules. The first module is **Information Fusion (IF)**, which transforms the raw narrative into a structured and verified baseline by aligning events with attributes and by annotating logical consistency. This step reduces dispersion, weak signals, and hidden relations in the text. The second module is **Query-driven Structured Reasoning (QSR)**, which treats missing or unclear premises as explicit obstacles. These obstacles are converted into queries and are resolved through hypothesis construction and state enrichment. This process continues until the state is sufficient to support a final conclusion. This modular structure supports both clarity and control during inference.

This design allows the model to make missing premises and latent causal gaps explicit, and to resolve them before committing to an answer. As a result, SABA reduces logical leaps, limits unsupported assumptions, and supports stable reasoning over long narratives. The reasoning process is also transparent, since each step updates an explicit state and records what information was added and why. This property supports later inspection and analysis. We evaluate SABA on a non-interactive detective benchmark called Detective Puzzle and on several general reasoning benchmarks. On the most difficult split of Detective Puzzle, SABA achieves a clear improvement over strong baselines in both answer correctness and evidence grounding.

Overall, our contributions are as follows. First, we focus on a form of non-interactive narrative reasoning where the main difficulty is truth reconstruction under information asymmetry. Second, we propose SABA, a framework that performs awareness before action through explicit state construction and obstacle resolution. Third, we introduce an evaluation setting that measures both final correctness and evidence use, which supports a more detailed analysis of reasoning behavior.

2 Related Work

Our work relates to three directions for improving complex reasoning in large language models, addressing a specific under-explored failure mode: in non-interactive long narratives, models often commit early based on incomplete premises and propagate this error.

2.1 The Challenge of Long-Context Narrative Reasoning

Unlike interactive settings where agents can query for missing information (Zhu et al., 2025; Wu et al., 2024), non-interactive narratives present fixed, diffuse evidence with distractors. This amplifies the lost-in-the-middle effect, causing early retrieval errors to compound (Dziri et al., 2023; Del and Fishel, 2023). Decomposition strategies (e.g., Least-to-Most (Zhou et al., 2023)) reduce local difficulty but do not ensure global coherence when clues are interdependent and scattered. The bottleneck is aligning dispersed evidence into a consistent, reusable situation model. SABA addresses this via an *Information Fusion* module, which consolidates the narrative into a unified, verifiable event-centric state before final deduction, making latent associations explicit and preventing evidence dilution.

2.2 Limitations of Post-Hoc Refinement

Methods like Self-Refine (Madaan et al., 2023) and Reflexion (Shinn et al., 2023) iteratively refine an initial output using self-feedback. However, such answer-then-correct paradigms are prone to confirmation bias (Huang et al., 2024; Stechly et al., 2023): subsequent steps rationalize the initial conclusion rather than re-audit its premises. This is severe in non-interactive narratives where a flawed early hypothesis dominates. SABA avoids this by shifting refinement from the candidate answer to the underlying knowledge state, enforcing an explicit audit for completeness and consistency before commitment.

2.3 From Unstructured Chains to Structured State Management

Works like Chain-of-Thought (Wei et al., 2022) and Tree-of-Thoughts (Yao et al., 2023) externalize reasoning traces but operate over unstructured text, lacking explicit representation of missing or inconsistent information. Recent advances in structured state tracking (Wang et al., 2024) suggest explicit state management improves reliability, yet state completion under missing premises remains open. SABA formalizes reasoning as iterative construction and verification of a structured state. It introduces *Query-driven Structured Reasoning* to treat missing/conflicting premises as active obstacles, transforming them into targeted queries for state repair—converting narrative reasoning into a deliberate process of filling a verified knowledge state.

Summary. Prior work separately addresses long-context challenges, iterative refinement, and structured traces. SABA synthesizes these into a state-first framework that builds and audits a coherent knowledge foundation before answer generation, mitigating premature commitment in non-interactive narrative reasoning.

3 Proposed Method

The core idea of SABA is to perform long-horizon narrative reasoning via an explicit recursive control loop that alternates between (i) structured information fusion and consistency annotation, and (ii) obstacle-driven query decomposition and hypothesis generation. This design explicitly represents missing premises and latent causal gaps, thereby reducing logical leaps and information forgetting in standard chain-of-thought reasoning. This recursive structure allows the system to repeatedly refine its internal state, which supports stable reasoning over long and complex narratives.

3.1 Information Fusion

Let $D = \{x_1, \ldots, x_n\}$ denote the raw narrative units. The goal of Information Fusion (IF) is to transform D into a structured and verified baseline state for subsequent reasoning. IF explicitly organizes and annotates narrative evidence to counteract dispersion, weak signaling, and implicit relations in long texts, which otherwise hinder reliable retrieval and reuse. This transformation ensures that later reasoning stages operate on an explicit and stable representation rather than on fragmented textual traces.

Event Alignment. We decompose D into a backbone sequence of core events

$$S = \{s_1, \ldots, s_m\}, \tag{1}$$

and a set of heterogeneous attributes

$$A = \{a_1, \ldots, a_p\}, \tag{2}$$

including actions, object states, locations, and evidentiary descriptors.

This decomposition separates the *narrative skeleton* (what happened and in what order) from the *descriptive details* (what properties, objects, or side conditions are involved), which are often scattered and weakly localized in the text. The alignment step aims to explicitly bind these descriptive attributes back to the events they qualify, thereby making implicit associations explicit and retrievable. This explicit binding reduces the need for later inference steps to reconstruct such links from memory alone. We then define an alignment mapping:

$$\Phi_{\text{map}} : A \to 2^S, \tag{3}$$

which assigns each attribute to one or more backbone events. This yields aligned units

$$d_i = \big(s_i, \{a \in A \mid s_i \in \Phi_{\text{map}}(a)\}\big), \tag{4}$$

and the aligned sequence $D_{\text{aligned}} = \{d_1, \ldots, d_m\}$.

Operationally, Φ_{map} is implemented by jointly considering semantic relevance, temporal proximity, and entity overlap between a and s. Allowing a multi-assignment ($A \to 2^S$) ensures that attributes that span multiple events (e.g., persistent object states or long-term intentions) are not artificially localized to a single point, which would otherwise reintroduce information loss. This design choice preserves long-range dependencies that are critical for coherent narrative interpretation.

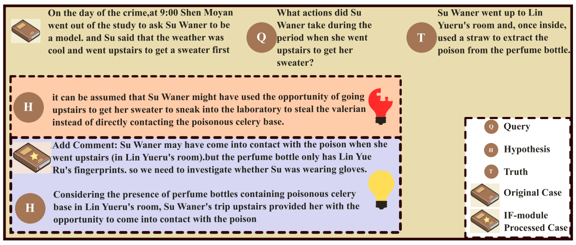
LConsistency Check. While alignment densifies information, it does not guarantee that the resulting structure is logically coherent. We therefore perform an explicit consistency annotation step to detect potential conflicts and uncertainties that may later mislead the reasoning process. This step complements alignment by adding a layer of logical validation.

For each aligned unit d_i, we compute a verification comment:

$$b_i = \psi_{\text{vfy}}(d_i, D_{\text{aligned}} \setminus d_i), \tag{5}$$

where ψ_{vfy} checks temporal, entity-state, and causal consistency. The result is the baseline state:

$$D_{\text{base}} = \{(d_i, b_i)\}_{i=1}^{m}. \tag{6}$$



On the day of the crime,at 9:00 Shen Moyan went out of the study to ask Su Waner to be a model. and Su said that the weather was cool and went upstairs to get a sweater first

What actions did Su Waner take during the period when she went upstairs to get her sweater?

Su Waner went up to Lin Yueru's room and, once inside, used a straw to extract the poison from the perfume bottle.

it can be assumed that Su Waner might have used the opportunity of going upstairs to get her sweater to sneak into the laboratory to steal the valerian instead of directly contacting the poisonous celery base.

Add Comment: Su Waner may have come into contact with the poison when she went upstairs (in Lin Yueru's room).but the perfume bottle only has Lin Yue Ru's fingerprints. so we need to investigate whether Su was wearing gloves.

Considering the presence of perfume bottles containing poisonous celery base in Lin Yueru's room, Su Waner's trip upstairs provided her with the opportunity to come into contact with the poison

Figure 2: Visualization of the IF module's impact. The pre-association of dispersed clues provides a verified cognitive baseline for subsequent reasoning.

Each comment b_i serves as a localized diagnostic signal indicating whether d_i is potentially problematic, underspecified, or contradictory relative to the rest of the narrative. These annotations are not used to discard information, but to mark it with explicit uncertainty or risk, which can then be explicitly handled by later reasoning stages instead of being silently ignored. This approach ensures that uncertainty remains visible and actionable throughout the reasoning process.

Empirical Analysis. As illustrated in Figure 2, standard LLMs often suffer from the "lost-in-the-middle" effect and clue dilution within long narratives. While traditional models may identify explicit individual clues, they frequently overlook implicit associations, for instance, failing to connect the action of "going upstairs" with the presence of "poison storage." The IF module establishes a verified cognitive baseline by pre-associating such dispersed attributes. By annotating latent risks (e.g., "potential contact with poison"), IF ensures that critical evidence remains highly available throughout the reasoning trajectory, effectively preventing the information forgetting prevalent in vanilla CoT. This empirical observation supports the functional role of IF in stabilizing long-horizon reasoning.

Takeaways 1. *Information Fusion converts an unstructured narrative into a dense, structured evidence representation. Feature Alignment makes implicit associations explicit and retrievable across reasoning steps. Consistency Annotation externalizes uncertainty and conflict, preventing silent propagation of errors.*

3.2 Query-driven Structured Reasoning

Given a task T (e.g., determining Motive, Modus Operandi, and Final Judgment), SABA performs recursive reasoning over the state D_t. The key idea of Query-driven Structured Reasoning (QSR) is to explicitly expose missing premises, resolve them via targeted queries, and incrementally enrich the reasoning state until a logically sufficient basis for T is reached. This process treats reasoning as a progressive construction of support rather than as a single inference step.

Gap Identification. Instead of directly attempting to infer T, the model first diagnoses what is missing. At iteration t, it identifies a set of reasoning obstacles

$$\Omega_t = \mathcal{M}(p_{\text{aware}} \mid D_t, T), \tag{7}$$

where each obstacle ω represents a missing or underspecified premise required to infer T from D_t. This step ensures that reasoning failures are addressed at their source rather than after they propagate.

Each obstacle is normalized as

$$\omega = (\tau(\omega), \mathrm{dim}(\omega), \mathrm{req}(\omega)), \tag{8}$$

where $\tau(\omega)$ is the type (e.g., MissingLink, Ambiguity, MotiveGap), $\mathrm{dim}(\omega)$ is the blocked task dimension, and $\mathrm{req}(\omega)$ is the missing requirement. This representation makes gaps in the reasoning process explicit and manipulable. It also enables systematic handling of different forms of incompleteness.

Each obstacle ω_i is then decomposed into subqueries

$$Q_{i,t} = \mathcal{M}(p_{\text{dec}} \mid \omega_i, D_t), \tag{9}$$

such that each $q \in Q_{i,t}$ targets a specific component of $\mathrm{req}(\omega_i)$. This decomposition converts abstract reasoning gaps into concrete information needs that can be individually addressed. As a result, each gap becomes a well-defined unit of further analysis.

Hypothesis Construction. For each sub-query q, a hypothesis is generated as

$$h = \mathcal{M}(p_{\text{hypo}} \mid q, D_t), \tag{10}$$

yielding $H_{i,t} = \{h\}$. These hypotheses act as tentative logical bridges that fill the detected gaps and are later validated through consistency and interaction with other evidence. They allow the system to explore possible explanations in a controlled manner. The reasoning state is updated as:

$$D_{t+1} = D_t \cup Q_t \cup H_t, \tag{11}$$

where $Q_t = \bigcup_i Q_{i,t}$ and $H_t = \bigcup_i H_{i,t}$. This update accumulates resolved premises while preserving the full trace of how each piece of information

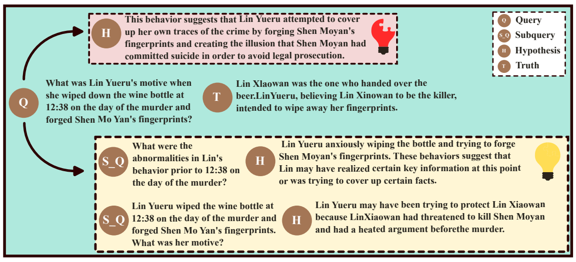


Figure 3: Visualization of the QSR module's impact: How query decomposition resolves logical obstacles and identifies hidden evidence.

was introduced. Such traceability supports both transparency and later error analysis.

The recursion terminates when either $\Omega_t = \emptyset$ (logical closure) or a maximum depth $t_{\max}$ is reached. The final conclusion is synthesized as

$$y = \mathcal{M}(p_{\text{syn}} \mid D_t, T). \tag{12}$$

This termination rule ensures that the process is both complete and computationally bounded.

Empirical Analysis. As shown in Figure 3, traditional CoT often incurs "logical leaps", leading to premature accusations based on superficial proximity. In contrast, QSR treats the absence of a causal link as a formal reasoning obstacle. Through recursive decomposition, it generates heuristic hypotheses (H) that successfully uncover the deep-seated motivation behind specific actions (e.g., "protecting Lin Xiaowan"). This behavior illustrates how QSR reduces unsupported inference while increasing causal depth.

Takeaways 2. *QSR reframes reasoning as iterative gap detection and resolution rather than direct deduction. Obstacle decomposition converts abstract missing premises into concrete, answerable queries. Recursive enrichment yields a transparent and logically grounded reasoning trajectory.*

3.3 Workflow of SABA

Algorithm 1 summarizes the complete SABA reasoning procedure. Given a raw narrative D and a target task T, the algorithm first constructs a structured and verified baseline via Information Fusion (Phase 1), and then iteratively refines this baseline via Query-driven Structured Reasoning (Phase 2). The core idea is to treat reasoning as progressive state completion: instead of directly inferring the answer, the model incrementally constructs the set of premises required to support a valid conclusion.

Specifically, the algorithm alternates between (i) diagnosing missing premises (obstacles), (ii) decomposing them into concrete queries, and (iii) generating hypotheses that fill these gaps. This loop continues until no critical obstacles remain or a maximum recursion depth is reached, at which point final synthesis is performed. This iterative structure ensures that reasoning proceeds in a controlled and verifiable manner.

Algorithm 1 SABA Recursive Reasoning

```
Require: Raw narrative $D$, Task $T$, Max depth $t_{max}$
Ensure: Final conclusion $y$
{Phase 1: Information Fusion (IF)}
$D_{aligned} \leftarrow \mathcal{M}(p_{align} \mid D)$ {Feature Alignment}
$D_{base} \leftarrow \{(d_i, \mathcal{M}(p_{vfy} \mid d_i)) \mid d_i \in D_{aligned}\}$ {Comment}

{Phase 2: Query-driven Structured Reasoning (QSR)}
$D_0 \leftarrow D_{base}, t \leftarrow 0$
while $t < t_{max}$ do
    $\Omega_t \leftarrow \mathcal{M}(p_{aware} \mid D_t, T)$ {Obstacle Identification}
    if $\Omega_t = \emptyset$ then
        break {Logical Closure}
    end if
    $Q_t, H_t \leftarrow \emptyset, \emptyset$
    for each obstacle $\omega_i \in \Omega_t$ do
        $Q_{i,t} \leftarrow \mathcal{M}(p_{dec} \mid \omega_i, D_t)$ {Query Decomposition}
        $H_{i,t} \leftarrow \{\mathcal{M}(p_{hypo} \mid q, D_t) \mid \forall q \in Q_{i,t}\}$ {Hypothesis Generation}
        $Q_t \leftarrow Q_t \cup Q_{i,t}, H_t \leftarrow H_t \cup H_{i,t}$
    end for
    $D_{t+1} \leftarrow D_t \cup \{Q_t, H_t\}$ {State Enrichment}
    $t \leftarrow t + 1$
end while

{Final Synthesis}
$y \leftarrow \mathcal{M}(p_{syn} \mid D_t, T)$ {Final Deduction}
return $y$
```

At each iteration t, the model logs the state

$$\mathcal{L}_t = (\Omega_t, Q_t, H_t, D_t), \tag{13}$$

which explicitly records what was missing, what was queried, what was hypothesized, and how the reasoning state evolved. This trace enables post-hoc inspection, error analysis, and verification of logical consistency. It also supports reproducibility by making the reasoning path observable.

Overall, the workflow ensures that information is introduced only when required to resolve a specific obstacle, which limits unnecessary speculation and constrains the propagation of unsupported assumptions. By transforming implicit chains of thought into explicit state transitions, SABA produces a transparent and auditable reasoning process. This design improves both reliability and interpretability in long-horizon narrative reasoning.

Category	Level	Samples	Avg. Length
Detective Puzzle	Easy	5	1050 words
	Medium	15	1150 words
	Complex	11	950 words
General	HotpotQA, StrategyQA, Big-Bench Hard		

Table 1: Benchmark statistics.

4 Experiments

4.1 Experimental Setup

Benchmarks. To evaluate SABA under varying degrees of information asymmetry, we adopt a complementary set of reasoning benchmarks. Dataset statistics are summarized in Table 1.

(1) Detective Puzzle (DP). Our primary benchmark for non-interactive endogenous truth reconstruction. It contains 31 cases with three difficulty levels: *Easy* (direct inference), *Medium* (multi-step causality), and *Complex* (implicit clues with red herrings). In our collection, 28 cases are sourced from the "5-Minute Mystery" platform, and the remaining 3 cases are adapted from classic detective novels.

(2) Multi-hop Reasoning. We include HotpotQA (Yang et al., 2018) to test evidence integration across disconnected passages, and StrategyQA (Geva et al., 2021) to assess bridging implicit commonsense gaps.

(3) Big-Bench Hard (BBH). We report the average accuracy over 23 challenging tasks from BBH (Suzgun et al., 2022) to verify whether SABA generalizes beyond detective-style deduction.

Evaluation Metrics. We evaluate whether a model reconstructs the truth via a coherent evidentiary chain rather than relying on superficial cues. For DP, we report both correctness and evidence-grounding metrics; for general benchmarks, we follow standard protocols. Specifically, for DP we report Suspect Accuracy (SA) for perpetrator identification, along with Motive Recall (R-M), Modus Operandi Recall (R-O), and Clue Coverage Rate (CCR). CCR measures the breadth of critical clue exploration.

To measure evidence grounding for motive and modus operandi, we compute semantic recall via atomic semantic matching: we decompose both prediction and reference into atomic propositions, compute embedding similarity, and apply greedy one-to-one matching with threshold $\tau = 0.5$.

For general benchmarks, we follow standard protocols. For HotpotQA, we report Answer EM (Ans) and Supporting Facts F1 (SF). For StrategyQA and BBH, we report accuracy.

Baselines. We compare SABA against three categories of prompting paradigms.

(1) Basic Reasoning: Direct prompting, Chain-of-Thought (CoT) (Wei et al., 2022), and Self-Consistency (SC) (Wang et al., 2023b).

(2) Decomposition and Planning: SELF-DISCOVER (Zhou et al., 2024), and Graph-of-Thought (GoT) (Besta et al., 2024).

(3) Iterative Optimization: Self-Refine (Madaan et al., 2023), CRITIC (Gou et al., 2024), and S^2R (Ma et al., 2025).

Implementation Details. We implement SABA using Deepseek-V3 and Gemini-1.5-Flash as backbone models. To ensure determinism and reproducibility, we set the decoding temperature to 0.0 for all experiments and use default API settings for other parameters. For semantic similarity computation, we use all-MiniLM-L6-v2 (Reimers and Gurevych, 2019). Average inference cost T is reported as a relative value compared to the Direct baseline. This normalized metric accounts for the substantial variations in input token counts across different benchmarks. Given the high cost of API-based inference, each configuration was evaluated using a single run. Accordingly, we report primary metrics in Tables 2, 3 and 4 without variance statistics.

4.2 Experimental Results

Comprehensive evaluation results are detailed in Table 2 (DeepSeek-V3) and Appendix Table 4 (Gemini-1.5-Flash). Across both models, SABA consistently achieves the strongest overall performance among compared methods, particularly on long-context complex reasoning tasks.

Performance on Detective Puzzle Benchmarks. On DeepSeek-V3, SABA achieves SA scores of 85.4, 82.1, and 78.1 on Easy, Medium, and Complex tasks, respectively. The largest gain is observed on the Complex set, where SABA improves SA from 70.1 (the best baseline, GoT) to 78.1 (+8.0). Meanwhile, SABA also improves evidence-grounding metrics on Complex, reaching CCR 83.8 (vs. 77.1 for the strongest baselines), R-M 73.1 (vs. 69.5 for GoT), and R-O 71.7 (vs. 67.4 for S^2R), indicating more thorough exploration of critical clues and more faithful reconstruction of motive

Table 2: Main results of **Deepseek-V3** across DP and general benchmarks. For DP, each cell reports results for three difficulty levels (*Easy* / *Medium* / *Complex*) from top to bottom. **HQA** denotes HotpotQA, **SQA** denotes StrategyQA. All results are shown in % except for inference costs T.

Method	**DP**				**HQA**		**SQA**	**BBH**	T
	SA	**R-M**	**R-O**	**CCR**	**Ans**	**SF**	**Acc**	**Acc**	
Direct	65.1 48.3 40.5	65.2 64.1 59.6	60.3 58.2 58.7	71.2 62.3 58.9	62.5	52.1	82.1	78.5	**1.0**
CoT	68.2 58.4 45.6	77.3 65.1 60.8	65.2 61.8 59.8	75.7 72.1 62.1	68.2	60.5	87.5	86.1	2.4
Self-Refine	70.7 65.7 55.3	67.1 66.4 62.2	68.3 65.2 61.1	80.4 76.1 65.6	68.4	59.2	88.2	87.5	6.2
SC(k=5)	72.1 70.1 62.9	68.5 64.7 61.8	69.4 66.2 62.5	81.3 80.2 70.9	72.1	62.1	90.2	89.4	12.1
CRITIC	77.3 73.8 66.1	68.1 65.3 64.7	70.6 68.7 63.2	88.4 87.8 73.2	74.1	68.5	90.5	87.8	29.2
S^2R	79.4 76.1 68.5	70.5 66.8 63.4	76.5 72.3 67.4	89.5 88.2 77.1	73.5	68.4	91.8	90.5	18.5
SELF-DISC.	80.8 75.3 68.6	72.4 68.1 64.5	78.3 70.0 65.1	87.2 86.8 75.7	75.4	68.4	92.3	91.2	4.8
GoT	84.2 76.7 70.1	78.3 80.2 69.5	82.4 72.3 67.2	90.3 88.1 77.1	78.1	73.2	91.8	90.5	35.5
SABA	**85.4** **82.1** **78.1**	**83.2** **75.3** **73.1**	**86.1** **73.1** **71.7**	**94.0** **93.1** **83.8**	**78.5**	**73.6**	**94.5**	**93.1**	25.8

and modus operandi under implicit gaps and red herrings.

In contrast, as task difficulty increases, linear prompting strategies such as CoT exhibit weaker evidence coverage on Complex (CCR 62.1). Optimization-based methods such as Self-Refine improve SA on Complex (55.3) but remain limited in evidence grounding (CCR 65.6), suggesting that iterative refinement may still overlook missing premises when the narrative contains strong distractors. Results on Gemini-1.5-Flash (Table 4) show a similar trend, demonstrating robustness across models of varying scales.

Generalization Across General Reasoning Benchmarks. To verify that SABA's advantage stems from general reasoning principles, we evaluated it on mainstream benchmarks. As shown in Table 2 (right), SABA improves HotpotQA Answer EM to 78.5 and Supporting Facts F1 to 73.6, slightly surpassing the strongest baseline GoT (78.1 / 73.2). On StrategyQA and BBH, SABA achieves accuracies of 94.5% and 93.1%, respectively, maintaining a clear lead over S^2R and SELF-DISCOVERY. These results suggest that treating missing premises as explicit obstacles to be resolved benefits both long-context narratives and multi-hop settings.

Inference Efficiency Analysis. We evaluate inference efficiency using the normalized cost T relative to the Direct baseline. While SABA incurs higher consumption ($T = 25.8$ on DeepSeek-V3) compared to CoT ($T = 2.4$), it is significantly more efficient than GoT ($T = 35.5$). Given the consistent gains in SA and evidence-grounding metrics (R-M/R-O/CCR), the extra computation is effectively devoted to resolving causal links and enforcing logical consistency, representing a favorable trade-off between reasoning depth and computational overhead compared to existing graph-based or iterative strategies.

Analysis of Results. As shown in Table 3, removing any component leads to a significant performance decay, validating their synergistic contribution. The most substantial degradation occurs when disabling obstacle identification (*w/o Awareness*),

Table 3: Ablation study on DP-Complex and StrategyQA using DeepSeek-V3. All results are shown in %.

Variant	**DP-Complex**		**StrategyQA**
	SA	**CCR**	**Accuracy**
SABA (Full)	**78.1**	**83.8**	**94.5**
— w/o IF	69.4	71.2	82.4
— w/o Awareness	61.5	62.5	76.8
CoT	45.6	62.1	87.5

where SA drops from 78.1 to 61.5 and StrategyQA accuracy decreases from 94.5 to 76.8. This collapse supports the importance of explicitly diagnosing knowledge gaps to prevent premature commitment under incomplete evidence. Removing the Information Fusion module (*w/o IF*) also reduces SA (78.1 → 69.4) and CCR (83.8 → 71.2), underscoring the necessity of consolidating fragmented narratives into a verified foundational state.

5 Conclusions and Limitations

In this paper, we introduced **SABA**, a reasoning paradigm based on the *Self-Awareness Before Action* principle that decouples cognitive assessment from answer generation and frames deduction as an iterative process of gap identification and resolution. This design treats reasoning as a controlled and stepwise process and aims to improve stability and reliability in complex tasks.

However, our study has several limitations. First, SABA depends on the self-evaluation ability of the backbone model, which can limit obstacle detection quality for smaller models. Second, its recursive process introduces higher latency despite prefix caching, which may affect real-time use and practical deployment. Third, due to the significant computational cost of API-based inference, all experiments were conducted with a single deterministic run under fixed settings, precluding a statistical analysis of variance across multiple runs. Additionally, the framework relies on the *IF module* for structured input processing, and a fully end-to-end extraction of clues remains an open problem that is important for future improvement.

In future work, the integration of *Retrieval-Augmented Generation (RAG)* could further improve factual grounding and reliability by providing external support for evidence and reducing residual errors caused by missing knowledge.

References

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. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(16):17682–17690.

Maksym Del and Mark Fishel. 2023. True detective: A deep abductive reasoning benchmark undoable for gpt-3 and challenging for gpt-4. *Preprint*, arXiv:2212.10114.

Nouha Dziri, Ximing Lu, Melanie Sclar, Xiang Lorraine Li, Liwei Jiang, Bill Yuchen Lin, Peter West, Chandra Bhagavatula, Ronan Le Bras, Jena D. Hwang, Soumya Sanyal, Sean Welleck, Xiang Ren, Allyson Ettinger, Zaid Harchaoui, and Yejin Choi. 2023. Faith and fate: Limits of transformers on compositionality. *Preprint*, arXiv:2305.18654.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Preprint*, arXiv:2101.02235.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. 2024. Critic: Large language models can self-correct with tool-interactive critiquing. *Preprint*, arXiv:2305.11738.

Jie Huang, Xinyun Chen, Swaroop Mishra, Huaixiu Steven Zheng, Adams Wei Yu, Xinying Song, and Denny Zhou. 2024. Large language models cannot self-correct reasoning yet. *Preprint*, arXiv:2310.01798.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12).

Tushar Khot, Harsh Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2023. Decomposed prompting: A modular approach for solving complex tasks. *Preprint*, arXiv:2210.02406.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts. *Preprint*, arXiv:2307.03172.

Ruotian Ma, Peisong Wang, Cheng Liu, Xingyan Liu, Jiaqi Chen, Bang Zhang, Xin Zhou, Nan Du, and Jia Li. 2025. S^2r: Teaching llms to self-verify and self-correct via reinforcement learning. *Preprint*, arXiv:2502.12853.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. *Preprint*, arXiv:2303.17651.

Piotr Piękos, Mateusz Malinowski, and Henryk Michalewski. 2021. Measuring and improving BERT's mathematical abilities by predicting the order of reasoning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 383–394, Online. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Noah Shinn, Federico Cassano, Edward Berman, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: Language agents with verbal reinforcement learning. *Preprint*, arXiv:2303.11366.

Zirui Song, Yuan Huang, Junchang Liu, Haozhe Luo, Chenxi Wang, Lang Gao, Zixiang Xu, Mingfei Han, Xiaojun Chang, and Xiuying Chen. 2025. Beyond survival: Evaluating llms in social deduction games with human-aligned strategies. *Preprint*, arXiv:2510.11389.

Kaya Stechly, Matthew Marquez, and Subbarao Kambhampati. 2023. Gpt-4 doesn't know it's wrong: An analysis of iterative prompting for reasoning problems. *Preprint*, arXiv:2310.12397.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. 2022. Challenging big-bench tasks and whether chain-of-thought can solve them. *Preprint*, arXiv:2210.09261.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel R. Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *Preprint*, arXiv:2305.04388.

Shenzhi Wang, Chang Liu, Zilong Zheng, Siyuan Qi, Shuo Chen, Qisen Yang, Andrew Zhao, Chaofei Wang, Shiji Song, and Gao Huang. 2023a. Avalon's game of thoughts: Battle against deception through recursive contemplation. *Preprint*, arXiv:2310.01320.

Xinyi Wang, Alfonso Amayuelas, Kexun Zhang, Liangming Pan, Wenhu Chen, and William Yang Wang. 2024. Understanding reasoning ability of language models from the perspective of reasoning paths aggregation. *Preprint*, arXiv:2402.03268.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023b. Self-consistency improves chain of thought reasoning in language models. *Preprint*, arXiv:2203.11171.

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 *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Dekun Wu, Haochen Shi, Zhiyuan Sun, and Bang Liu. 2024. Deciphering digital detectives: Understanding LLM behaviors and capabilities in multi-agent mystery games. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 8225–8291, Bangkok, Thailand. Association for Computational Linguistics.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *Preprint*, arXiv:1809.09600.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. *Preprint*, arXiv:2305.10601.

Jintian Zhang, Xin Xu, Ningyu Zhang, Ruibo Liu, Bryan Hooi, and Shumin Deng. 2024. Exploring collaboration mechanisms for llm agents: A social psychology view. *Preprint*, arXiv:2310.02124.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V Le, and Ed H. Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations*.

Pei Zhou, Jay Pujara, Xiang Ren, Xinyun Chen, Heng-Tze Cheng, Quoc V. Le, Ed H. Chi, Denny Zhou, Swaroop Mishra, and Huaixiu Steven Zheng. 2024. Self-discover: Large language models self-compose reasoning structures. *Preprint*, arXiv:2402.03620.

Qinglin Zhu, Runcong Zhao, Bin Liang, Jinhua Du, Lin Gui, and Yulan He. 2025. Player*: Enhancing llm-based multi-agent communication and interaction in murder mystery games. *Preprint*, arXiv:2404.17662.

A Additional Experimental Results

In this section, we provide the detailed performance of the Gemini-1.5-Flash backbone across all benchmarks.

Table 4: Main results of **Gemini-1.5-Flash** across DP and general benchmarks. For DP, each cell reports results for three difficulty levels (*Easy* / *Medium* / *Complex*) from top to bottom. **HQA** denotes HotpotQA, **SQA** denotes StrategyQA. All results are shown in % except for inference costs T.

Method	**DP**				**HQA**		**SQA**	**BBH**	T
	SA	**R-M**	**R-O**	**CCR**	**Ans**	**SF**	**Acc**	**Acc**	
Direct	40.2 35.8 32.1	48.4 44.2 42.7	47.3 45.1 38.2	45.1 35.4 32.1	52.4	42.1	72.4	68.5	**1.0**
CoT	42.7 38.2 35.4	52.3 45.6 45.1	51.5 48.2 40.4	49.2 48.4 35.2	58.1	48.5	78.1	75.2	2.4
Self-Refine	42.1 38.2 32.4	50.2 48.5 45.1	49.9 44.2 42.9	52.4 42.1 38.5	55.2	46.4	79.4	76.1	6.2
SC(k=5)	52.0 51.5 42.1	48.5 45.3 42.4	50.4 45.2 42.1	65.4 58.4 45.5	60.1	50.2	82.1	79.8	12.1
CRITIC	58.3 52.2 50.2	52.1 48.5 55.4	65.2 52.1 48.2	72.1 68.2 60.1	62.4	55.1	68.4	62.1	29.2
S^2R	65.4 62.1 67.2	58.2 55.4 62.1	67.2 60.1 55.4	75.1 72.4 65.2	62.4	55.1	83.4	81.2	18.5
SELF-DISC	62.8 60.2 56.4	56.4 52.1 58.2	69.3 62.9 52.1	74.2 70.1 62.5	65.2	57.4	85.2	83.1	4.8
GoT	66.8 64.3 60.7	59.1 56.1 62.4	70.3 62.4 58.2	78.4 75.2 65.4	68.2	61.2	84.1	82.5	35.5
SABA	**74.1** **72.5** **68.5**	**65.2** **68.5** **72.4**	**72.4** **65.2** **62.1**	**85.1** **83.2** **75.4**	**70.1**	**63.2**	**87.2**	**86.4**	16.8