NEURO-SYMBOLIC REASONING FRAMEWORK FOR LONG-DOCUMENT UNDERSTANDING

Anonymous authors
Paper under double-blind review

ABSTRACT

Multimodal long-document understanding remains challenging because relevant evidence is sparsely distributed across lengthy and redundant documents. Existing retrieval-based methods optimize page-level relevance only, yet individually relevant pages may still fail to collectively provide all the evidence required by a question. Moreover, cross-page reasoning is often generated without explicit verification, allowing unsupported steps and logical gaps to remain hidden under incomplete evidence. We therefore formulate long-document understanding as a neuro-symbolic reasoning process that couples neural evidence discovery with symbolic verification. Neural models derive question-specific evidence requirements, localize complementary evidence across pages, and construct a structured reasoning hypothesis. Symbolic constraints evaluate evidence coverage, validate intermediate reasoning, and enforce answer conditions. Failed checks trigger a bounded repair loop to retrieve missing evidence and revise the reasoning hypothesis, while unresolved cases lead to abstention. Experiments on three multimodal long-document benchmarks demonstrate state-of-the-art performance, improved evidence localization and token efficiency, and interpretable evidence navigation with auditable verification traces.

1 INTRODUCTION

With the rapid progress of vision-language models, multimodal document understanding has attracted increasing attention Mathew et al. (2021); Tanaka et al. (2021); Masry et al. (2022). Real-world documents, however, are often lengthy, information-dense, and highly redundant. Answering questions over such documents requires models to move beyond local visual-textual recognition. They must navigate heterogeneous pages, identify complementary evidence, and reason across text, tables, figures, and layouts Ma et al. (2024); Deng et al. (2025).

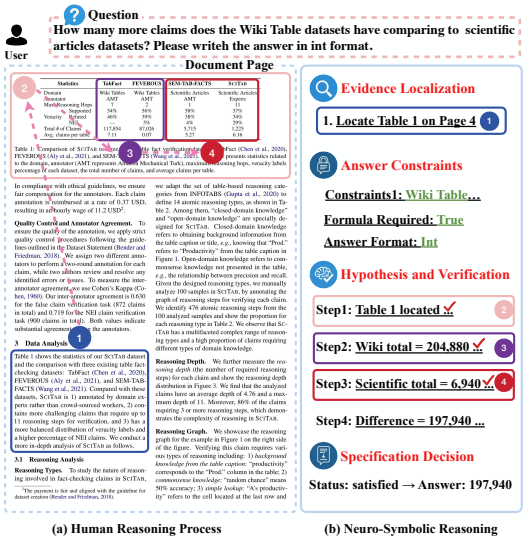

Figure 1: (a) Humans progressively locate evidence. (b) Our framework combines localization, specification, and verification.

Existing methods address these challenges through document retrieval, page selection, multimodal encoding, and long-context modeling Cho et al. (2024); Jain et al. (2025); Wu et al. (2025); Liu et al. (2026). Most follow a retrieval-based paradigm: they identify relevant pages or regions and then generate an answer from the retrieved context. While this paradigm reduces irrelevant information, it mainly optimizes page-level relevance. However, individually relevant pages may still fail to collectively provide all the evidence required by a question. For example, answering a question with multiple conditions requires evidence for each condition, which may be distributed across different pages and expressed with varying degrees of lexical over-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

lap. Page-level retrieval may capture the most explicit match while missing another logically necessary piece of evidence. The retrieved context is therefore relevant but incomplete.

Moreover, sufficient evidence does not guarantee valid reasoning. A model may still ignore the required scope, rely on unsupported evidence, or apply an incorrect operation. Without explicit verification, these errors can be hidden behind a plausible answer. Long-document understanding therefore requires both complete evidence and verified reasoning.

Neuro-symbolic reasoning provides a natural foundation for addressing these challenges. It combines neural document interpretation with explicit question-derived constraints. These constraints make evidence coverage and reasoning validity checkable, enabling targeted retrieval of missing evidence and rejection of unsupported conclusions. Prior work has used structured programs and reasoning representations for symbolic execution, logical checking, and evidence grounding Ping et al. (2025); Park et al. (2026); Feng et al. (2025); Zhao et al. (2026).

However, most existing neuro-symbolic approaches assume that the required input evidence has already been provided. Multimodal long-document understanding introduces an additional difficulty: the supporting evidence must first be discovered from many heterogeneous pages. This creates a strong dependency between evidence acquisition and reasoning verification. Localization determines what can be verified, while verification reveals whether the localized evidence is complete. These stages prevent failures in reasoning from being translated into concrete evidence needs.

Our key insight is that evidence localization and reasoning verification should be governed by shared symbolic constraints derived from the question. These constraints serve two complementary roles. Evidence requirements specify what information must be found across the document, turning localization from relevance ranking into coverage-aware evidence acquisition. Answer constraints specify what evidence, reasoning steps, and output conditions must be satisfied before an answer can be accepted. When a check fails because a concrete piece of evidence is missing, the same constraint provides a targeted signal for further navigation.

Based on this insight, we propose a two-stage neuro-symbolic framework for multimodal long-document understanding. In the localization stage, neural models interpret the document and derive question-specific evidence requirements. These symbolic requirements guide page selection, encouraging the acquisition of complementary rather than redundant evidence. In the reasoning stage, a neural model constructs a structured hypothesis over the localized evidence, while an explicit answer specification evaluates evidence coverage, intermediate reasoning steps, and final answer conditions. Failed checks with identifiable evidence gaps trigger a bounded repair loop that retrieves additional evidence and revises the reasoning hypothesis. When sufficient support cannot be established, the framework abstains instead of producing an unsupported answer.

Experiments on three multimodal long-document benchmarks demonstrate state-of-the-art performance. Our framework also improves evidence localization and token efficiency, while producing interpretable evidence navigation and auditable verification traces. Our contributions are summarized as follows:

- We identify a fundamental gap in retrieval-based long-document understanding: page-level relevance does not guarantee collective evidence sufficiency, while sufficient evidence alone does not guarantee valid reasoning. We formulate these two requirements within a unified neuro-symbolic reasoning framework.
- We introduce question-derived symbolic evidence requirements that guide coverage-aware localization, enabling neural models to acquire complementary evidence distributed across heterogeneous pages.
- We develop specification-guided hypothesis verification that evaluates evidence support, intermediate reasoning, and answer conditions. Failed checks guide bounded evidence repair, while unresolved cases lead to abstention.

2 Related Work

Multimodal Document Question Answering. Document question answering is a core task in multimodal document understanding. Early benchmarks and models mainly focus on single-page or

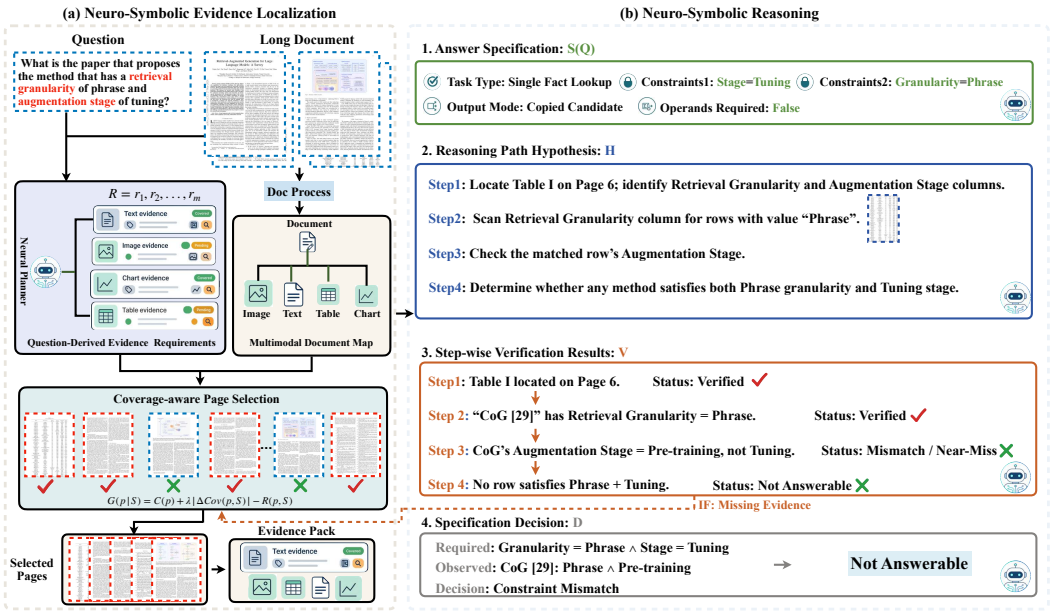

Figure 2: Overview of our neuro-symbolic framework. Neural agents interpret multimodal document pages, derive evidence requirements, and construct reasoning hypotheses. The symbolic component encodes evidence requirements, answer specifications, verification states, and loop signals to guide evidence acquisition and downstream reasoning.

single-image documents, requiring systems to understand text, layout, and visual structures jointly. Representative datasets include DocVQA Mathew et al. (2021), InfographicVQA Tanaka et al. (2021), and ChartQA Masry et al. (2022), while models such as LayoutLM Xu et al. (2020) and Donut Kim et al. (2022) improve document representation through layout-aware pre-training or OCR-free modeling. However, these studies usually assume that the relevant evidence lies within a limited visual context. Recent work extends document QA to long multimodal documents, where evidence may be sparse, dispersed, and mixed with substantial redundancy, as reflected in benchmarks such as MMLongBench Ma et al. (2024) and LongDocURL Deng et al. (2025). Existing methods tackle these challenges through multimodal retrieval, page selection, and iterative refinement Cho et al. (2024); Jain et al. (2025); Wu et al. (2025). Recent agentic frameworks further decompose long-document understanding into navigation, retrieval, and refinement steps Han et al. (2025); Zhu et al. (2026); Zhang et al. (2026); Chen et al. (2026); Liu et al. (2026). Although these methods improve evidence access and multi-step interaction, they provide limited modeling of the question-derived evidence requirements and answer constraints needed for fully supported answers.

Neuro-symbolic Reasoning and Verification. Neuro-symbolic frameworks combine neural perception with structured representations for compositional and verifiable reasoning Manhaeve et al. (2018); Dong et al. (2019); Badreddine et al. (2022). Recent methods formalize multimodal inputs for symbolic execution Ping et al. (2025); Kamali & Kordjamshidi (2025), validate generated reasoning through logical consistency and evidence grounding Feng et al. (2025); Zhao et al. (2026), or integrate verification and hypothesis refinement into visual and embodied reasoning Park et al. (2026); Choi et al. (2025). These approaches improve reliability and interpretability, but operate on a given input or reasoning trace. Long-document understanding requires identifying evidence across heterogeneous pages before verification. Our method extends neuro-symbolic control to both evidence localization and answer reasoning. It derives symbolic evidence requirements for coverage-aware page selection and an answer specification for hypothesis verification. Thus, question-derived constraints govern collective evidence coverage, reasoning validity, and answerability.

3 METHODS

Problem Setup. Let $D = \{p_i\}_{i=1}^{N}$ denote a document of N pages, where p_i is the page at index i, and let Q denote a natural-language question. The goal is to return an answer A supported by D. MinerU2.5-Pro Wang et al. (2026) converts D into a reusable document map memory $\mathcal{M}$. We

factor inference into localization and reasoning. The localization function f_{loc} derives evidence requirements $\mathcal{R}$ from Q and selects an initial page set $P^{(0)}$:

$$(\mathcal{R}, P^{(0)}) = f_{\text{loc}}(D, Q). \tag{1}$$

The selected pages are converted by an assembly function $E(\cdot)$ into an evidence pack. The verification function f_{ver} then checks a structured reasoning hypothesis under a question-derived answer specification $\mathcal{S}(Q)$:

$$(H, V) = f_{\text{ver}}\big(Q, E(P^{(0)}), \mathcal{S}(Q)\big). \tag{2}$$

Here, H denotes the reasoning hypothesis and V denotes the verification state.

In this work, *symbolic* refers to explicit, machine-readable structures derived from Q, including evidence requirements, answer constraints, reasoning steps, verification states, and repair signals. These structures guide page localization, hypothesis verification, and evidence loop repair.

3.1 Neuro-Symbolic Localization

Document Map Memory. We parse each document once with MinerU2.5-Pro before localization. For each page p_i, we store its image, OCR text, layout elements with page indices and bounding boxes, and a compact visual caption when visual elements are present:

$$\mathcal{M}[p_i] = (I_i, T_i, L_i, U_i), \tag{3}$$

where I_i is the page image, T_i is the OCR text, L_i denotes layout elements, and U_i denotes the visual caption. The full document memory is $\mathcal{M} = \{\mathcal{M}[p_i]\}_{i=1}^{N}$. It supplies coarse cues for page navigation and source content for evidence assembly, allowing question-derived evidence requirements to be matched against textual, visual, and layout information.

Evidence Requirements. A whole-question navigator may focus on the most salient clue and miss complementary evidence. We therefore derive from Q a set of atomic evidence requirements $\mathcal{R} = \{r_j\}_{j=1}^{m}$. Each requirement is

$$r_j = (n_j, d_j, \tau_j, \mathcal{Q}_j), \tag{4}$$

where n_j names the requirement, d_j describes the needed evidence, τ_j denotes its modality or semantic type, and $\mathcal{Q}_j$ contains search queries. For example, a comparison yields one requirement for each operand, and a numerical question yields requirements for the values needed in the computation. These explicit requirements turn page selection into coverage-aware localization: the system selects pages that collectively satisfy the question's evidence needs, rather than relying only on whole-question relevance.

Candidate Navigation. The navigator searches $\mathcal{M}$ for pages that may satisfy the evidence requirements $\mathcal{R}$. A memory-based matcher scores pages using textual, visual, and structural cues from the requirement queries. In parallel, a multimodal neural navigator examines candidate pages and produces page-level votes. Explicit page references in Q are resolved to page indices and forced into the candidate pool. This yields two complementary signals for each page: a memory-matching score and a neural navigation score.

Page Selection. Let $R_{\text{mem}}(p)$ and $R_{\text{nav}}(p)$ denote the memory-matching and neural navigation scores of page p. We normalize them and compute

$$C(p) = \alpha \widehat{R}_{\text{mem}}(p) + (1 - \alpha)\widehat{R}_{\text{nav}}(p), \tag{5}$$

where $\alpha \in [0, 1]$ balances memory matching and neural navigation. High relevance alone does not ensure evidence coverage. We therefore select pages with a requirement-coverage objective:

$$p^{\star} = \arg\max_{p \in \mathcal{P} \setminus P} \left[C(p) + \lambda |\Delta\text{Cov}(p; \mathcal{R}, P)| - \rho(p, P) \right]. \tag{6}$$

Here, $\Delta\text{Cov}(p; \mathcal{R}, P)$ denotes the requirements covered by p but not by the current set P, λ rewards new coverage, and $\rho(p, P)$ penalizes redundancy with selected pages. Selection stops when the page budget K is reached or no candidate has positive gain. Resolved explicit references are then merged into the selected set. Thus, neural scores estimate page relevance, while symbolic requirements enforce evidence coverage, allowing the system to select a page that completes a missing requirement rather than the top-ranked pages.

Algorithm 1 Neuro-Symbolic Evidence Localization

Require: document D; question Q; page budget K
Ensure: pages $\mathcal{P}$; memory $\mathcal{M}$; evidence requirements $\mathcal{R}$; initial pages $P^{(0)}$

$\mathcal{P}, \mathcal{M} \leftarrow \mathrm{ParseAndRepresent}(D)$
$\mathcal{R} \leftarrow \mathrm{DeriveEvidenceRequirements}(Q)$
for all $p \in \mathcal{P}$ **do**
  $R_{\mathrm{mem}}(p) \leftarrow \mathrm{RequirementMatch}(p, \mathcal{R}, \mathcal{M})$
  $R_{\mathrm{nav}}(p) \leftarrow \mathrm{NavigationVotes}(p, Q, \mathcal{R}, \mathcal{M})$
  $C(p) \leftarrow \alpha \widehat{R}_{\mathrm{mem}}(p) + (1-\alpha)\widehat{R}_{\mathrm{nav}}(p)$
end for
$P^{(0)} \leftarrow \emptyset$
while $|P^{(0)}| < K$ and the best marginal gain is positive **do**
  $p^\star \leftarrow \arg\max_{p \notin P^{(0)}} C(p) + \lambda |\Delta \mathrm{Cov}(p)| - \rho(p, P^{(0)})$
  $P^{(0)} \leftarrow P^{(0)} \cup \{p^\star\}$
end while
$P^{(0)} \leftarrow \mathrm{MergeExplicitReferences}(P^{(0)}, Q)$

return $\mathcal{P}, \mathcal{M}, \mathcal{R}, P^{(0)}$

3.2 Neuro-Symbolic Reasoning

Evidence Pack. Let P denote the page set returned by localization. For each $p_i \in P$, the element localizer crops question-relevant regions from L_i and stores them as $\mathcal{Z}_i$. The evidence assembler then builds

$$E(P) = \{(p_i, I_i, T_i, \mathcal{Z}_i) \mid p_i \in P\}. \tag{7}$$

The pack preserves page identity, full-page layout, OCR text, and fine-grained visual regions.

Answer Specification. Before verification, the verifier derives an answer specification $\mathcal{S}(Q)$ from the original question. This specification is the symbolic form of the answer constraints introduced above. We write

$$\mathcal{S}(Q) = (\kappa, \eta, \mathcal{C}, \mathcal{O}, \omega), \tag{8}$$

where κ is the expected answer type, η specifies output conditions such as cardinality and format, $\mathcal{C}$ is the set of critical constraints from Q, $\mathcal{O}$ records required operations such as counting, comparison, or numerical computation, and ω decides whether the evidence is sufficient. For example, a list question may forbid missing or extra items, and a numerical question must support its operands and operation. Thus, $\mathcal{R}$ guides what to retrieve, while $\mathcal{S}(Q)$ governs what can be accepted as an answer.

Reasoning Hypothesis. Given the evidence pack $E(P)$, a neural hypothesizer constructs a structured reasoning hypothesis before final answer generation. Let $H = \{h_k\}_{k=1}^{K_H}$ denote the hypothesis with K_H ordered steps:

$$h_k = (e_k, o_k, y_k, \nu_k, \delta_k). \tag{9}$$

Here, e_k specifies the evidence to inspect, o_k denotes the operation, y_k is the expected finding, ν_k specifies the verification method, and δ_k lists prerequisite steps. Operations include lookup, reading, counting, comparison, aggregation, and calculation. Verification methods include text matching, full-page inspection, crop-based visual inspection, cross-page lookup, and numerical checking. This structure makes each intermediate claim checkable by linking it to evidence, operation, verification method, and dependencies

Independent Verification. The independent verifier receives the same evidence pack as the hypothesizer, together with Q, H, and $\mathcal{S}(Q)$. It first checks the constraints in $\mathcal{C}$ against the original question and evidence. It then verifies each step h_k. First, the target evidence must be present. Second, the operation o_k must match the evidence type and the question constraint. Finally, the observed result must support the expected finding y_k. The verifier may revisit a full page, inspect a crop, search OCR text, or execute a numerical check. Related evidence is not sufficient when it refers to the wrong entity, period, page, or visual element.

Algorithm 2 Neuro-Symbolic Reasoning

Require: question Q; pages $\mathcal{P}$; memory $\mathcal{M}$; initial pages $P^{(0)}$; maximum rounds T
Ensure: answer A; hypothesis H; verification state V

$E^{(0)} \leftarrow \mathrm{AssembleEvidence}(Q, \mathcal{P}, P^{(0)}, \mathcal{M})$
$H^{(0)} \leftarrow \mathrm{HypothesizePath}(Q, E^{(0)})$
$\mathcal{S}, V^{(0)} \leftarrow \mathrm{SpecifyAndVerify}(Q, H^{(0)}, E^{(0)})$
$\bar{X} \leftarrow (P^{(0)}, E^{(0)}, H^{(0)}, V^{(0)})$
for $t = 0$ to $T - 1$ **do**
  if $V^{(t)}$ is verified **then**
    break
  end if
  $g^{(t)} \leftarrow \Gamma(Q, V^{(t)}, E^{(t)})$
  if $g^{(t)}$ is not actionable **then**
    break
  end if
  $\pi^{(t)} \leftarrow \Pi(g^{(t)}, P^{(t)}, \mathcal{P})$
  $P^{(t+1)}, Z^{(t+1)} \leftarrow \mathrm{ExecuteRepair}(\pi^{(t)})$
  $E^{(t+1)} \leftarrow \mathrm{Assemble}(Q, P^{(t+1)}, Z^{(t+1)})$
  $H^{(t+1)} \leftarrow \mathrm{HypothesizePath}(Q, E^{(t+1)})$
  $\mathcal{S}, V^{(t+1)} \leftarrow \mathrm{SpecifyAndVerify}(Q, H^{(t+1)}, E^{(t+1)})$
  if $\mathrm{AcceptRepair}(V^{(t)}, V^{(t+1)}, g^{(t)})$ **then**
    $\bar{X} \leftarrow (P^{(t+1)}, E^{(t+1)}, H^{(t+1)}, V^{(t+1)})$
  end if
end for
$(\bar{P}, \bar{E}, \bar{H}, \bar{V}) \leftarrow \bar{X}$
if $\mathrm{SpecificationSatisfied}(\bar{V})$ **then**
  $A \leftarrow \mathrm{GenerateAnswer}(Q, \bar{E}, \bar{H}, \bar{V})$
else
  $A \leftarrow \mathrm{Abstain}(\bar{V})$
end if

return $A, \bar{H}, \bar{V}$

Agreement among neural samples does not guarantee evidence support, because candidates may share the same missing evidence or perceptual error. The verifier therefore bases acceptance on explicit constraints and independently observed document evidence. Its output is

$$V = (V_{\mathrm{con}}, V_{\mathrm{step}}, V_{\mathrm{spec}}, \mathcal{G}, s), \tag{10}$$

where V_{con} records the results for question constraints in $\mathcal{C}$, V_{step} records the status of each hypothesis step, and V_{spec} states whether the support rule ω is satisfied. The set $\mathcal{G}$ contains concrete evidence gaps, and $s \in \{\textit{verified}, \textit{partially verified}, \textit{not verified}\}$ is the overall status. This state separates missing evidence from invalid reasoning: only a concrete missing target should trigger further document navigation. Neural models thus propose how to interpret the document, while the symbolic state determines whether that interpretation is admissible.

Evidence Loop. At feedback round t, let $P^{(t)}$, $E^{(t)}$, $H^{(t)}$, and $V^{(t)}$ denote the current pages, evidence pack, hypothesis, and verification state. The loop is triggered only when $\mathcal{G}$ contains a concrete evidence gap. The gap is converted into a repair signal. It specifies the missing target, search scope, and action, such as page addition, OCR extraction, cross-page search, or deterministic counting. The framework then updates the evidence, rebuilds $E^{(t+1)}$ and $H^{(t+1)}$, and applies the same independent verifier. A new state is retained only when it resolves the stated gap or improves verification without invalidating previously supported constraints. The loop terminates when s is *verified*, no actionable gap remains, or the budget of T rounds is exhausted. The final answer is generated only if V_{spec} satisfies ω.

4 EXPERIMENTS

4.1 EXPERIMENTAL SETUP

Datasets. We evaluate our method on three long-context document benchmarks: MMLongBench Ma et al. (2024), DocBench Zou et al. (2025), and LongDocURL Deng et al. (2025). MMLongBench targets multimodal long-document understanding over lengthy PDF documents, where answers may depend on text, tables, page layouts, and other document elements. DocBench evaluates document question answering on visually rich pages and emphasizes layout-aware understanding and fine-grained reasoning. LongDocURL extends the evaluation to URL-based long-context information seeking, where models need to identify evidence from long and noisy web-style documents. These benchmarks cover document sources, evidence types, and reasoning challenges.

Evaluation Protocol. We mainly evaluate the overall pipeline performance on MMLongBench and DocBench, and report the final answer semantic correctness. These two datasets cover a wide range of question types. They also include *Not Answerable* questions, whose answers are not supported by the document. Such questions allow us to evaluate whether a model produces hallucinated answers when the evidence is insufficient. Since DocBench does not provide evidence-page annotations, we evaluate evidence-page localization on MMLongBench and LongDocURL, both of which provide supporting-page labels.

Metrics. For MMLongBench and DocBench, we follow an evaluation protocol similar to the official MMLongBench setting. We use Qwen3.6-Plus as an LLM judge to determine whether the predicted answer and the ground-truth answer express the same semantic information. The judge assigns a binary score to each sample, and we report the average score as answer semantic correctness. For evidence-page localization, we report page recall and full recall following prior long-document QA studies. Page recall measures the fraction of gold evidence pages that are covered by the predicted pages. Full recall measures whether all gold evidence pages are retrieved for a question. These metrics evaluate both partial evidence coverage and complete evidence recovery.

Baselines. We organize the compared methods into three groups. First, we include general-purpose VLMs that directly take document page images as input, such as GPT-4o, Claude-4-Sonnet, Gemini-2.5-Flash, Qwen3.6-Plus, and Gemini-2.5-Pro. Second, we compare with RAG-based long-document methods, including M3DocRAG Cho et al. (2024) and MoLoRAG Wu et al. (2025). These methods retrieve document evidence before answer generation and are strong references for evidence-centered long-document QA. Third, we evaluate prompt-based agentic methods, including MDocAgent Han et al. (2025), SLEUTH Liu et al. (2026), DocDancer Zhang et al. (2026), and DocLens Zhu et al. (2026). These methods use multi-step prompting, tool use, or agent-style reasoning to handle complex document questions. We use published results under identical benchmark settings when available and reproduce methods otherwise.

4.2 OVERALL PERFORMANCE

Results on MMLongBench. Table 1 presents results on MMLongBench. Our method achieves the highest overall accuracy of 73.0%, outperforming the reproduced DocLens baseline with Qwen3.6-Plus by 5.0 points. We obtain the best results across all question types. The gains on TAB, CHA, and FIG reflect the benefit of our neuro-symbolic design. Symbolic evidence requirements guide page selection, and step-wise verification checks the relevant tables, charts, and figures. The large gain on UNA shows that constriaint-guided reasoning helps reject unsupported answers. These results support our main claim that explicit evidence requirements and answer specifications improve both evidence grounding and final answer correctness.

Results on DocBench. Table 2 shows the results on DocBench. Our method reaches 86.3% overall accuracy, exceeding the reproduced DocLens baseline by 2.9 points. On the UNA subset, it improves from 85.3% to 90.1%. This result is important because UNA questions require the system to identify insufficient support rather than generate a plausible answer. The improvement supports the effectiveness of answer constraints and independent verification.

Table 1: Results on MMLongBench Ma et al. (2024). Accuracy (%, higher is better). Best results in each column are shaded in dark. † denotes our own reproduced results.

Method	MMLongBench Ma et al. (2024)						
	TXT	LAY	CHA	TAB	FIG	UNA	ALL
VLMs							
GPT-4o	46.3	46.0	45.3	50.0	44.1	20.2	42.8
Claude-4-Sonnet	50.4	49.4	50.5	57.3	43.9	59.0	53.4
Gemini-2.5-Flash	44.0	53.2	46.0	43.9	48.2	56.7	49.6
Qwen3.6-plus	63.5	66.2	65.7	41.2	63.2	40.3	57.2
Gemini-2.5-Pro	52.1	62.1	55.5	55.3	54.0	59.9	58.1
Claude-4-Sonnet w/ OCR	52.7	51.6	50.0	58.1	45.3	65.9	56.0
Gemini-2.5-Flash w/ OCR	55.9	54.9	52.7	63.4	50.3	60.8	58.5
Gemini-2.5-Pro w/ OCR	59.7	65.3	60.8	68.3	55.7	58.4	63.3
VLM-based Agentic Frameworks							
M3DocRAG Cho et al. (2024) (w/ Qwen2-VL-7B)	30.0	23.5	18.9	20.1	20.8	5.8	21.0
MDocAgent Han et al. (2025) (w/ GPT-4o)	–	–	–	–	–	–	42.0
M3DocRAG Cho et al. (2024)	54.63	53.52	53.94	44.15	41.51	48.64	45.90
MoLoRAG Wu et al. (2025)	54.86	54.93	54.85	46.89	46.41	51.67	48.75
MDocAgent Han et al. (2025)	53.81	56.39	52.97	45.98	45.52	49.71	47.82
MACT Yu et al. (2026) (w/ MiMo-VL-7B)	–	–	–	–	–	–	47.4
SLEUTH Liu et al. (2026) (w/ Qwen3VL-8B)	59.26	53.52	53.27	47.55	50.27	67.38	52.77
DocDancer Zhang et al. (2026) (w/ Gemini-2.5-Pro)	–	–	–	–	–	–	65.9
DocLens Zhu et al. (2026) (w/ Gemini-2.5-Pro)	63.7	64.6	64.3	69.7	60.2	72.2	67.6
DocLens Zhu et al. (2026) (w/ Qwen3.6-plus) †	70.0	64.2	62.5	68.7	64.8	62.9	68.0
Ours (w/ Qwen3.6-plus)	**71.4**	**68.1**	**67.7**	**74.4**	**66.8**	**77.3**	**73.0**

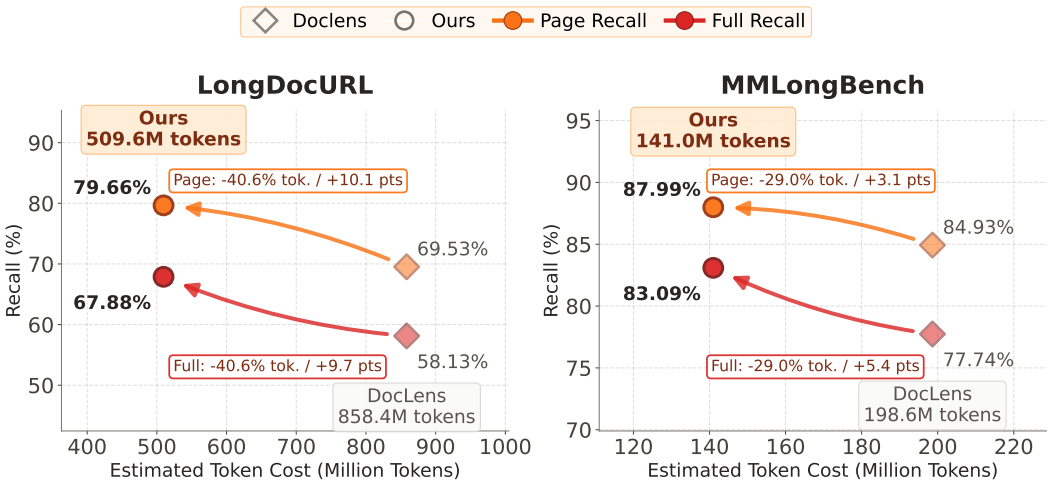

Figure 3: Comparison of evidence page retrieval between DocLens and our method. Our method obtains higher mean page recall and full recall with lower token consumption.

Evidence-page Localization. Figure 3 compares our method with DocLens on evidence-page retrieval. On LongDocURL, our method improves page recall from 69.53% to 79.66% and full recall from 58.13% to 67.88%, while reducing the estimated token cost from 858.4M to 509.6M. On MMLongBench, our method improves page recall from 84.93% to 87.99% and full recall from 77.74% to 83.09%, with a lower token cost of 141.0M compared with 198.6M. These results verify that evidence requirements improve coverage and reduce unnecessary page exploration.

Case Study. Figure 4 illustrates how our method solves a multimodal long-document question through neuro-symbolic reasoning. The question asks for signatures on page 15 and page 16. Our method localizes the two target pages and constructs an answer specification that restricts the scope to these pages and requires an integer-valued output. It verifies each step with grounded evidence: one signature is found on page 15 and one signature is found on page 16. After summing the

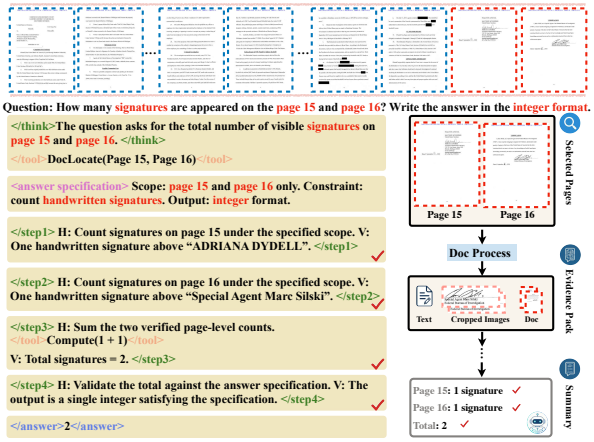

Figure 4: Case study of our neuro-symbolic reasoning for a counting question over specified document pages.

Method	DocBench	
	UNA	ALL
VLMs		
GPT-4o	–	63.1
Qwen3.6-plus	78.2	75.4
Gemini-2.5-Pro	77.0	73.1
VLM-based Agentic Frameworks		
MoLoRAG Wu et al. (2025)	79.2	73.1
MDocAgent Han et al. (2025)	73.9	71.1
MACT Yu et al. (2026)		
w/ MiMo-VL-7B	69.4	65.1
DocDancer Zhang et al. (2026)		
w/ Gemini-2.5-Pro	–	79.9
DocLens Zhu et al. (2026)†		
w/ Qwen3.6-plus	85.3	83.4
Ours(w/ Qwen3.6-plus)	**90.1**	**86.3**

Table 2: Results on DocBench Zou et al. (2025). Accuracy (%).

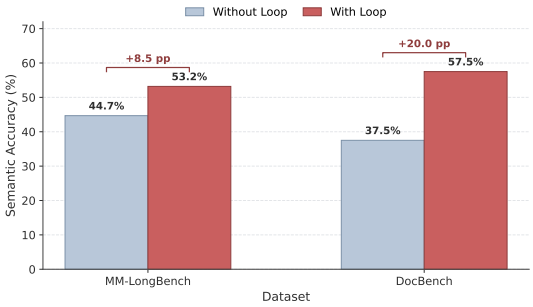

Figure 5: Accuracy with the evidence loop disabled vs. enabled on MMLongBench and DocBench.

Method	Accuracy (%)
w/o Evidence Loop	72.4
w/o Neuro-Symbolic Evidence Localization	71.7
w/o Neuro-Symbolic Reasoning	68.8
w/o All Components	68.0
Full Method	**73.0**

Table 3: Ablation of framework components on MMLongBench Ma et al. (2024).

verified page-level counts, the model outputs the final answer "2". This example shows how answer specifications turn a counting question into explicit scope and output constraints, and how step-wise verification grounds the final answer in page-level evidence.

4.3 Ablation Study

Effect of the Evidence Loop. Figure 5 shows the effect of enabling the evidence loop on samples that require loop repair. On MMLongBench, the semantic accuracy increases from 44.7% to 53.2%, yielding an absolute gain of 8.5 points. On DocBench, the improvement is larger, rising from 37.5% to 57.5%, with a gain of 20.0 points. These results indicate that iterative evidence checking is important for correcting uncertain or unsupported answers.

Component Ablation. Table 3 studies the contribution of each component on MMLongBench. The w/o Evidence Loop setting keeps Neuro-Symbolic Evidence Localization and Neuro-Symbolic Reasoning, but does not revisit samples with missing evidence pages. Removing the evidence loop, Neuro-Symbolic Evidence Localization, and Neuro-Symbolic Reasoning decreases accuracy to 72.4%, 71.7%, and 68.8%, respectively. Removing all components gives the lowest score of 68.0%. These drops align with our main contributions. Evidence requirements improve page coverage, answer constraints and step-wise verification reduce unsupported reasoning, and the bounded repair loop recovers missing evidence after verification failures. The full method achieves 73.0%, showing that these three contributions provide complementary gains.

5 Conclusion

We have presented a neuro-symbolic reasoning framework for reliable long-document understanding. Neural agents interpret heterogeneous document content, while symbolic constraints govern evidence localization and answer reasoning. Explicit evidence-sufficiency verification mitigates unsupported reasoning when the document lacks adequate support. Experiments demonstrate strong performance with interpretable and auditable verification traces.

AI USE STATEMENT

In this work, we used generative AI tools only for language polishing of the manuscript and for assisting with code debugging during system development. We have not used generative AI tools for idea generation, method design, experimental execution, or result analysis. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

This work evaluates exclusively on publicly released research benchmarks (MMLongBench, DocBench, and LongDocURL) and does not involve human subjects, personal or sensitive data, or proprietary documents. All datasets are used under their original research licenses. The proposed method targets document understanding and question answering; we do not foresee dual-use risks beyond those common to general document QA systems.

REPRODUCIBILITY STATEMENT

The full pipeline is specified in Section 3, including Algorithms 1 and 2, and the evaluation protocol is described in Section 4.1. Implementation details, prompts, and hyperparameters are provided in the supplementary material, which also contains the configurations needed to rerun all experiments. Reproduced baselines are marked with † in the result tables. Code will be released upon acceptance.

REFERENCES

Samy Badreddine, Artur d'Avila Garcez, Luciano Serafini, and Michael Spranger. Logic tensor networks. *Artificial Intelligence*, 303:103649, 2022.

Kaifeng Chen, Hongtao Liu, Qiyao Peng, Jian Yang, Yongqiang Liu, Xiaochen Zhang, and Qing Yang. Mardoc: A memory-aware refinement agent framework for multimodal long document qa. *arXiv preprint arXiv:2606.05749*, 2026.

Jaemin Cho, Debanjan Mahata, Ozan Irsoy, Yujie He, and Mohit Bansal. M3docrag: Multi-modal retrieval is what you need for multi-page multi-document understanding. *arXiv preprint arXiv:2411.04952*, 2024.

Wonje Choi, Jinwoo Park, Sang Hyun Ahn, Daehee Lee, and Honguk Woo. Nesyc: A neuro-symbolic continual learner for complex embodied tasks in open domains. *arXiv preprint arXiv:2503.00870*, 2025.

Chao Deng, Jiale Yuan, Pi Bu, Peijie Wang, Zhong-Zhi Li, Jian Xu, Xiao-Hui Li, Yuan Gao, Jun Song, Bo Zheng, et al. Longdocurl: a comprehensive multimodal long document benchmark integrating understanding, reasoning, and locating. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1135–1159, 2025.

Honghua Dong, Jiayuan Mao, Tian Lin, Chong Wang, Lihong Li, and Denny Zhou. Neural logic machines. In *International Conference on Learning Representations*, 2019.

Yu Feng, Nathaniel Weir, Kaj Bostrom, Sam Bayless, Darion Cassel, Sapana Chaudhary, Benjamin Kiesl-Reiter, and Huzefa Rangwala. Vericot: Neuro-symbolic chain-of-thought validation via logical consistency checks, 2025. URL https://arxiv.org/abs/2511.04662.

Siwei Han, Peng Xia, Ruiyi Zhang, Tong Sun, Yun Li, Hongtu Zhu, and Huaxiu Yao. Mdocagent: A multi-modal multi-agent framework for document understanding. *arXiv preprint arXiv:2503.13964*, 2025.

Chelsi Jain, Yiran Wu, Yifan Zeng, Jiale Liu, Shengyu Dai, Zhenwen Shao, Qingyun Wu, and Huazheng Wang. Simpledoc: Multi-modal document understanding with dual-cue page retrieval and iterative refinement. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 28398–28415, 2025.

Danial Kamali and Parisa Kordjamshidi. Neptune: A neuro-pythonic framework for tunable compositional reasoning on vision-language. *arXiv preprint arXiv:2509.25757*, 2025.

Geewook Kim, Teakgyu Hong, Moonbin Yim, JeongYeon Park, Jinyeong Yim, Wonseok Hwang, Sangdoo Yun, Dongyoon Han, and Seunghyun Park. Ocr-free document understanding transformer. In *European Conference on Computer Vision*, pp. 498–517, 2022.

Keliang Liu, Zizhi Chen, Mingcheng Li, Jingqun Tang, Dingkang Yang, and Lihua Zhang. Resolving evidence sparsity: Agentic context engineering for long-document understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19452–19462, 2026.

Yubo Ma, Yuhang Zang, Liangyu Chen, Meiqi Chen, Yizhu Jiao, Xinze Li, Xinyuan Lu, Ziyu Liu, Yan Ma, Xiaoyi Dong, et al. Mmlongbench-doc: Benchmarking long-context document understanding with visualizations. *Advances in Neural Information Processing Systems*, 37:95963–96010, 2024.

Robin Manhaeve, Sebastijan Dumancic, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. Deepproblog: Neural probabilistic logic programming. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 2263–2279, 2022.

Minesh Mathew, Dimosthenis Karatzas, and C. V. Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2200–2209, 2021.

Hyejin Park, Junhyuk Kwon, Suha Kwak, and Jungseul Ok. Viro: Robust and efficient neuro-symbolic reasoning with verification for referring expression comprehension, 2026. URL https://arxiv.org/abs/2601.12781.

Bowen Ping, Minnan Luo, Zhuohang Dang, Chenxi Wang, and Chengyou Jia. Autogps: Automated geometry problem solving via multimodal formalization and deductive reasoning, 2025. URL https://arxiv.org/abs/2505.23381.

Ryota Tanaka, Kyosuke Nishida, Sen Yoshida, Hiroya Takamura, and Mohit Bansal. Infographicvqa. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1697–1706, 2021.

Bin Wang, Tianyao He, Linke Ouyang, Fan Wu, Zhiyuan Zhao, Tao Chu, Yuan Qu, Zhenjiang Jin, Weijun Zeng, Ziyang Miao, et al. MinerU2.5-Pro: Pushing the limits of data-centric document parsing at scale. *arXiv preprint arXiv:2604.04771*, 2026.

Xixi Wu, Yanchao Tan, Nan Hou, Ruiyang Zhang, and Hong Cheng. Molorag: Bootstrapping document understanding via multi-modal logic-aware retrieval. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 14035–14056, 2025.

Yiheng Xu, Minghao Li, Lei Cui, Shaohan Huang, Furu Wei, and Ming Zhou. Layoutlm: Pre-training of text and layout for document image understanding. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1192–1200, 2020.

Xinlei Yu, Chengming Xu, Zhangquan Chen, Yudong Zhang, Shilin Lu, Cheng Yang, Jiangning Zhang, Shuicheng Yan, and Xiaobin Hu. Visual document understanding and reasoning: A multi-agent collaboration framework with agent-wise adaptive test-time scaling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12300–12311, 2026.

Qintong Zhang, Xinjie Lv, Jialong Wu, Baixuan Li, Zhengwei Tao, Guochen Yan, Huanyao Zhang, Bin Wang, Jiahao Xu, Haitao Mi, et al. Docdancer: Towards agentic document-grounded information seeking. *arXiv preprint arXiv:2601.05163*, 2026.

Guangzhen Zhao, Dechang Kong, Tongyu Wu, and Zhenjiang Dong. TrustTable: A neuro-symbolic auditing framework for faithful table QA. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 34907–34934, San Diego, California, United States, July 2026. Association for Computational Linguistics. doi: 10.18653/v1/2026.acl-long.1612. URL https://aclanthology.org/2026.acl-long.1612/.

Dawei Zhu, Rui Meng, Jiefeng Chen, Sujian Li, Tomas Pfister, and Jinsung Yoon. Doclens: A tool-augmented multi-agent framework for long visual document understanding. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 26804–26829, 2026.

Anni Zou, Wenhao Yu, Hongming Zhang, Kaixin Ma, Deng Cai, Zhuosheng Zhang, Hai Zhao, and Dong Yu. Docbench: A benchmark for evaluating llm-based document reading systems. In *Proceedings of the 4th International Workshop on Knowledge-Augmented Methods for Natural Language Processing*, pp. 359–373, 2025.

APPENDIX

A ADDITIONAL RESULTS ON LONGDOCURL

Table A1 supplements the main-paper localization analysis with final-answer accuracy on LongDocURL Deng et al. (2025). The table reports answer accuracy separately from the page-recall, full-recall, and token-cost analysis in the main paper. Our method achieves 68.10% accuracy, outperforming the reproduced DocLens result with the same Qwen3.6-plus backbone by 1.73 percentage points. It also exceeds SLEUTH, the strongest other agentic framework in the table, by 8.14 points. The same-backbone improvement is consistent with the claim that improved evidence acquisition and verification can translate into higher final-answer accuracy on long and noisy URL-based documents.

Table A1: Results on LongDocURL Deng et al. (2025). Accuracy (%, higher is better). The best result is shaded in dark orange. † denotes our reproduced results.

Method	**Accuracy (%)**
VLMs	
GPT-4o†	35.70
Qwen3.6-plus†	55.40
Gemini-2.5-Pro†	56.20
VLM-based Agentic Frameworks	
M3DocRAG Cho et al. (2024)	54.59
MoLoRAG Wu et al. (2025)	57.57
MDocAgent Han et al. (2025)	53.11
SLEUTH Liu et al. (2026)	59.96
DocLens Zhu et al. (2026) (w/ Qwen3.6-plus)†	66.37
Ours (w/ Qwen3.6-plus)†	**68.10**

B CASE STUDY: EVIDENCE-LOOP REPAIR

Figure A1 illustrates how our bounded evidence-repair loop operates on a MMLongBench counting example. The case demonstrates that our framework does not accept an answer based on partially sufficient evidence. Instead, question-derived constraints guide verification of the required scope, expose the missing page as an actionable evidence gap, and trigger a targeted evidence update before final reasoning.

C NEURO-SYMBOLIC DESIGN

We use "symbolic" to denote typed, machine-readable states, finite document operators, and explicit control rules rather than a general-purpose theorem prover. Neural modules perform open-ended document perception and semantic interpretation; symbolic structures specify required evidence, answer conditions, verification outcomes, repair actions, and admissible state transitions. The framework is therefore not merely a collection of prompts: neural proposals are consumed by an explicit verification-and-control interface.

Symbolic states. The framework connects the objects introduced in the main paper: evidence requirements $\mathcal{R}$ describe what must be located; answer specification $\mathcal{S}(Q)$ defines expected answer type, output conditions, critical constraints, required operations, and the support rule; reasoning hypothesis H contains ordered and checkable steps; verification state V records constraint, step, and specification status; evidence gaps $\mathcal{G}$ identify actionable missing targets; and repair plan π selects a bounded evidence update.

Finite operators. The implementation instantiates the main-paper operations with a finite operator set: `SELECT_PAGES`, `ASSEMBLE_EVIDENCE`, `PLAN_REASONING_PATH`, `VERIFY_QUESTIO`

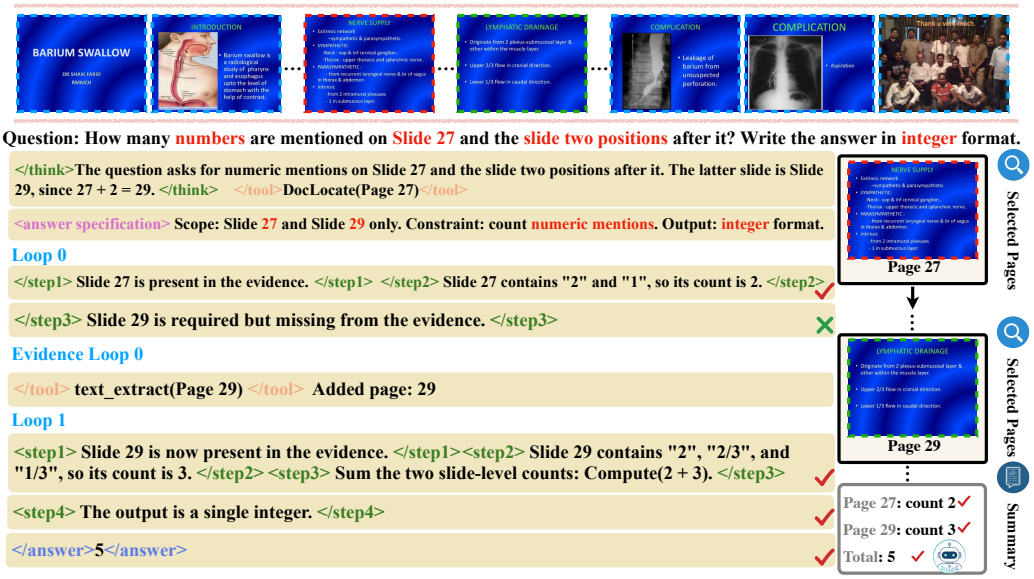

Figure A1: Case study of verification-driven evidence repair.

Stage	Neural role	Symbolic role
Localization	Estimate page relevance and interpret multimodal cues	Represent evidence requirements and enforce coverage-aware selection
Reasoning	Propose evidence-grounded reasoning steps	Record operations, dependencies, criticality, and answer specifications
Verification	Inspect text, layouts, tables, figures, and crops	Check question constraints, reasoning steps, and answer conditions
Repair	Propose missing-evidence targets	Restrict updates to finite reasons, actions, and budgets
Decision	Generate candidate answers	Accept, repair, or abstain according to verification state

Table A2: Neural and symbolic roles in the framework.

N_CONSTRAINTS, VERIFY_ANSWER_CONTRACT, VERIFY_REASONING_STEP, EXECUTE_DOCUMENT_TOOL, PLAN_REPAIR, and ACCEPT_OR_ABSTAIN. These implementation labels correspond to localization, evidence assembly, hypothesis construction, verification, repair, and final specification checking in Algorithms 1–2 of the main paper. Document tools have typed arguments and explicit scope. Metadata lookup, OCR retrieval, and counting execute deterministically over the parsed document representation; crop interpretation and semantic support checks remain neural and may be marked uncertain.

State transition. At round t, let $X^{(t)} = (P^{(t)}, E^{(t)}, H^{(t)}, V^{(t)}, \mathcal{S}(Q))$. The controller applies

$$\text{Next}(X^{(t)}) = \begin{cases} \textsc{Accept}, & \text{Sat}(V^{(t)}, \mathcal{S}(Q)), \\ \textsc{Repair}(g), & g \in \mathcal{G}^{(t)} \text{ is actionable}, \\ \textsc{Abstain}, & \text{otherwise.} \end{cases} \tag{A1}$$

Here, Sat is the supplementary notation for V_{spec} satisfying the support rule ω in the main paper. It requires all critical question constraints, reasoning steps, and applicable answer conditions to be supported. Repair is not a free-form retry: actions are selected from registered page-update, OCR, retrieval, counting, layout, visual-inspection, and reference-resolution operators. Consistent with Algorithm 2, a repaired state is retained only when it resolves the target gap or improves verification without invalidating previously supported constraints.

D IMPLEMENTATION DETAILS

Document representation. Each document is parsed once before question answering. For page p_i, the system stores the page image I_i, OCR text T_i, parsed layout elements L_i with element types and bounding boxes, and a compact visual caption U_i when visual elements are present. Following the main paper, $\mathcal{M}[p_i] = (I_i, T_i, L_i, U_i)$ and $\mathcal{M} = \{\mathcal{M}[p_i]\}_{i=1}^{N}$. Page identity is retained during retrieval and evidence assembly so that a later verification result can be traced to its source page or region.

Evidence requirement schema. Following the notation in the main paper, each requirement is represented as $r_j = (n_j, d_j, \tau_j, \mathcal{Q}_j)$, where n_j is the requirement name, d_j is the requested evidence, τ_j is its modality or semantic type, and $\mathcal{Q}_j$ is a set of search queries. A page contributes coverage when its textual, visual, or layout information supports the corresponding requirement. Explicit page references are resolved to page indices and merged into the candidate evidence set rather than treated only as ordinary relevance cues.

Evidence selection. The initial evidence set combines memory-based matching, neural navigation scores, explicit references, and requirement coverage. As defined in the main paper, the combined relevance score is $C(p) = \alpha \widehat{R}_{\mathrm{mem}}(p) + (1-\alpha)\widehat{R}_{\mathrm{nav}}(p)$, and selection rewards newly covered requirements while penalizing redundancy under the page budget K. The evidence pack follows $E(P) = \{(p_i, I_i, T_i, \mathcal{Z}_i) \mid p_i \in P\}$, where $\mathcal{Z}_i$ contains localized question-relevant regions. This representation preserves source pages while distinguishing missing evidence from present but inconsistent evidence.

Reasoning and verification records. Following the main paper, a reasoning step is stored as $h_k = (e_k, o_k, y_k, \nu_k, \delta_k)$, representing its target evidence, operation, expected finding, verification method, and dependencies. Step criticality is retained as an implementation field used by verification, but is not added to the mathematical tuple. The verifier returns $V = (V_{\mathrm{con}}, V_{\mathrm{step}}, V_{\mathrm{spec}}, \mathcal{G}, s)$, where $s \in \{\textit{verified}, \textit{partially verified}, \textit{not verified}\}$. Constraint- and step-level records additionally retain discrepancy notes and evidence references. Original-question constraints are checked before reasoning steps and answer-specification conditions.

Tool-call protocol. Every document-tool call contains an operator name, typed arguments, page or region scope, and a returned observation. Calls are restricted to the registered operator set and are recorded in the trace. Text lookup and counting operate over the stored document representation; visual inspection returns a neural observation that must be evaluated by the verifier. Tool results are therefore linked to the evidence state rather than appended as unsupported natural-language claims.

Repair protocol. A repair is triggered only by an actionable evidence gap, such as a missing page, referenced element, cross-page occurrence, statistic, or visual candidate. The repair controller selects a registered action and updates the evidence pack under the page budget and the maximum T repair rounds defined in the main paper. After each update, the reasoning hypothesis is rebuilt and the same independent verifier produces a new verification state rather than carrying forward an unchecked conclusion. The loop stops when the state is verified, no actionable gap remains, or the repair budget is exhausted. The best retained state is then passed to the final specification check; the system abstains when V_{spec} does not satisfy ω.

Answer contract enforcement. The answer specification is question-derived and follows $\mathcal{S}(Q) = (\kappa, \eta, \mathcal{C}, \mathcal{O}, \omega)$ from the main paper: expected answer type, output conditions, critical constraints, required operations, and the support rule. It conditions hypothesis verification and the final answer modules. Type, cardinality, scope, required slots, and operations are checked before acceptance. Missing critical support, disallowed extra items, wrong scope, or unsupported numerical operations prevent acceptance. Thus, agreement among sampled answers is insufficient when the evidence state violates a critical answer condition.

Trace contents. A complete trace records the question-derived requirements, selected pages, evidence references, reasoning steps, answer specification, constraint results, step results, tool calls,

repair actions, final state, and acceptance decision. These fields expose the interaction between neural proposals and symbolic control and support per-sample diagnosis of localization, verification, and repair failures. Trace records are auxiliary audit outputs; the inference behavior is defined by the state transitions and acceptance conditions described above.

E PROMPT DESIGN

Prompts elicit neural proposals and evidence inspections; they are not the symbolic layer by themselves. The gray boxes below reproduce the prompt excerpts used by the corresponding modules. The symbolic layer consists of the typed states, finite operators, verification records, bounded repair controller, and acceptance rule described above.

E.1 REASONING-PATH HYPOTHESIZER PROMPT

ROLE
You are a reasoning path planner for multi−page document question answering.
Given a question and a set of evidence pages from a long document, your task is to hypothesize the reasoning path that a human expert would follow to answer the question.

CRITICAL
You are NOT answering the question yourself −− you are planning HOW to verify the answer. The verifier will independently check each step using tools.

You must preserve the original wording and constraints of the question. Do NOT repair, rewrite, broaden, or reinterpret the question to make it answerable. Required constraints include the requested entity/object/activity, page or figure reference, model/dataset/context, time period, subgroup, and all operands needed for comparison or computation. Near−miss evidence is evidence for unanswerability, not a substitute answer path.

Instructions
For each step, specify step_id, description, target_page, target_element_type, target_element_bbox, operation, observed_evidence, expected_finding, verification_method, depends_on, cross_page, and is_critical.

Output Format
Return a single valid JSON object (no markdown code blocks, no extra text) with question_type, answer_contract, reasoning_steps, alternative_paths,
distinguishing_clue, refusal_criteria, and in_evidence.

E.2 VERIFIER PROMPT

ROLE
You are a reasoning path verifier for document question answering. You will verify whether each step of a hypothesized reasoning path can be confirmed by the available document evidence.

You must also verify the answer contract inferred from the question. The contract is not a dataset rule; it is a question−driven expectation about answer type, cardinality, required slots, candidate−set constraints, numeric operations, scope restrictions, and answerability conditions.

Instructions

0. First verify the original question constraints
Before evaluating the hypothesized reasoning steps, extract the exact constraints from the original question and verify whether the evidence satisfies them. Constraints include named entities, requested objects, page/figure/table references, dates, sections, comparison operands, and requested context qualifiers.

Do NOT repair or reinterpret the question. If the evidence only contains a near−miss, related item, broader category, different page/figure, or different context, mark that constraint as failed.

1. For each reasoning step:
Observe the step's observed_evidence and expected_finding, identify the SPECIFIC TARGET,
examine the provided page images, OCR text, and crops yourself, use tools when needed, and set status to verified, partially_verified, or not_verified.

Important Rules
The original question constraints have priority over the hypothesized path. A near−miss (similar name, adjacent value, related−but−different object) is NOT a find.

E.3 ANSWER SAMPLING PROMPT

ROLE:
You are an expert AI assistant specializing in multimodal long document understanding. Your task is to carefully analyze the provided page images (which may contain text, figures, tables or other content) and provide a precise answer to the user's question. Treat the provided pages as candidate evidence, not as guaranteed sufficient or guaranteed correct evidence.

You may be given an answer contract inferred from the question. Treat it as a hard constraint on what counts as a sufficient final answer. The contract is not dataset−specific; it summarizes whether the answer should be single−fact, list, closed−set selection, numeric, multi−slot, or answerability−sensitive, and whether extra items, missing slots, paraphrases, or near−miss substitutions are allowed.

Follow these instructions carefully:
− Core Objective: Your primary goal is to accurately and concisely answer the user's question based on the content of the provided document pages while strictly preserving the original question constraints.
− Rule of faithfulness: Be faithful. If the provided pages do not contain sufficient information to answer the user's question, you should answer 'Not answerable'.
− Exact−constraint rule: Do not repair the user's question by assuming typos, synonyms, broader categories, or intended alternatives.

Output Format:
Your entire response MUST be a single, valid JSON object and nothing else. The JSON object must contain exactly two fields: analysis and prediction.

E.4 ADJUDICATOR PROMPT

ROLE:
You are an expert AI assistant specializing in multimodal long document understanding. Your primary role is to serve as an aggregator of different answers provided by multiple AI agents for a given
question about a complex long document.

Critical Change from Original Adjudicator:
Unlike the original adjudicator that could NOT see the document, you now have DIRECT access to the document evidence (page images, OCR text, tables, charts). You must use this evidence to INDEPENDENTLY verify whether each candidate's answer is consistent with the document.

Follow these instructions carefully:
− Core Objective: Determine which answer is most appropriate by evaluating both the candidate analyses AND the actual document evidence provided to you.
− Independent Verification: You MUST cross−check candidate answers against the document evidence.
− Avoid Frequency Bias: Ignore the frequency with which an answer appears. Judge based on factual evidence from the document, not on consensus.

Using Verification Results (if provided):
IMPORTANT: You must read constraint_results first, then the per−step step_results (each step's status and discrepancy_note), NOT only the overall_status. The constraint and per−step findings are the authoritative signal −− overall_status alone is a coarse summary.