Logic Matters in Lightweight Hallucination Classification for RAG System

Ningyuan Yang
Duke Kunshan University
ningyuan.yang@dukekunshan.edu.cn

Kaizhu Huang*
Duke Kunshan University
kaizhu.huang@dukekunshan.edu.cn

Abstract

We propose a lightweight, modular framework for hallucination detection in Retrieval-Augmented Generation (RAG) systems, addressing the critical challenge where logical dependencies span across fragmented retrieval results. Through graph-based *semantic evidence aggregation*, which captures the implicit logical structure by clustering semantically coherent segments across retrieved documents via betweenness centrality, our approach enables small NLI models to handle multi-hop reasoning without task-specific training. We present two deployment configurations: a resource-efficient variant ($\approx$0.5B parameters) achieving 82.4% accuracy on HotPotQA-Derived at 85 ms latency, outperforming all sub-1B baselines by over 30%; and a higher-accuracy variant ($\approx$1.5B parameters) reaching 85.6%, surpassing 11B TrueTeacher while being 7$\times$ smaller and 1.7$\times$ faster. Experiments with six NLI discriminator models show consistent gains of $+6.7\%$–$+29.9\%$, confirming that graph-based evidence aggregation is NLI-agnostic and the primary performance driver. We also contribute **HotPotQA-Derived**, a new multi-hop hallucination benchmark preserving separate retrieved documents for systematic evaluation.

1 Introduction

Retrieval-Augmented Generation (RAG) (Lewis et al., 2020) has emerged as a powerful strategy for mitigating hallucinations in large language models (LLMs) by grounding their outputs in externally retrieved documents. However, under constrained computational budgets, the retrieval stage itself can introduce new hallucinations: for instance, when asked "Which Japanese city served as the imperial capital during the Heian period?", RAG may retrieve passages describing Kyoto's Heian Shrine and ancient palace grounds, yet the LLM hallucinates "Tokyo" and describes the Meiji-era Imperial Palace. Such errors arise because a compact model may lose track of precise information in retrieved segments or misinterpret their relationships. Specifically, the logical chain required to verify a claim

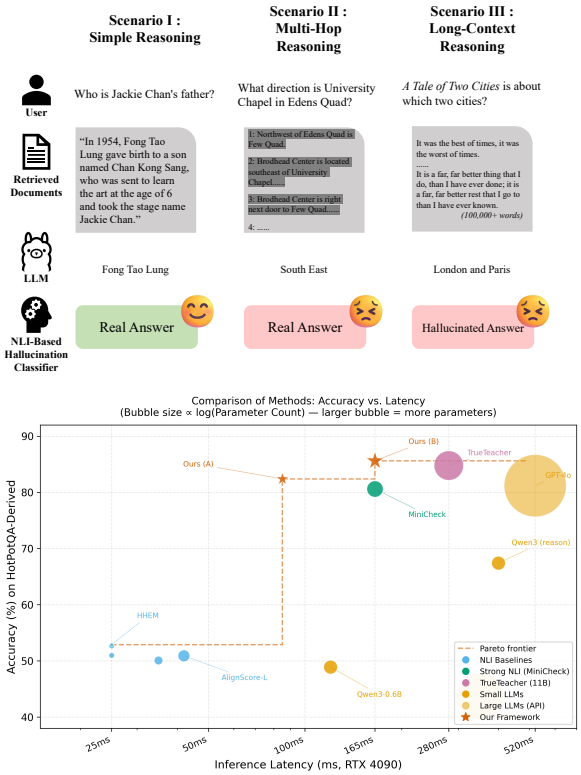

Figure 1: **Upper:** Dilemmas of NLI-based models in multi-hop and long-context hallucination detection. When evidence for a factual claim is scattered across separate passages, evaluating each document independently yields low NLI scores (0.03 and 0.12 individually), while combining them recovers the correct entailment score (0.63). **Lower:** Comparison of our framework with baselines on accuracy vs. inference latency. Bubble area encodes log(parameter count). Both configurations (starred) occupy the Pareto frontier, with Config A achieving $> 30\%$ accuracy gain over sub-1B baselines at 85 ms.

*Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1605–1617

often spans multiple disjoint passages. When such cross-document logic is fragmented, lightweight detectors fail to reconstruct the inferential dependencies, leading to undetected hallucinations.

Hallucination detection in RAG systems follows two main paths. The first adapts the language model through fine-tuning on hallucination-annotated corpora, yielding strong in-domain performance but requiring large domain-specific annotated datasets. The second applies a Natural Language Inference (NLI) model to score factual consistency. While this incurs minimal overhead, its effectiveness degrades sharply on tasks requiring long-context understanding or multi-hop reasoning (Figure 1): the restricted parameter size of NLI models prevents them from holistically perceiving evidence distributed across multiple documents.

Prior work has focused primarily on alleviating long-context challenges. RAGAS (Es et al., 2025) and Provenance (Sankararaman et al., 2024) reduce per-call context length by evaluating each retrieved document independently. Grounded Context Retrieval (Gerner et al., 2025) further segments and filters individual passages. Despite these advances, the complexity of *aggregating semantically related evidence* across multiple retrieved documents remains elusive; existing approaches still struggle to detect hallucinations in multi-hop reasoning settings.

Our Approach. We introduce a compact, three-module framework designed for multi-hop hallucination detection under tight compute constraints. Our method addresses the limited reasoning capacity of small models by explicitly clustering semantically coherent evidence across documents before consistency scoring:

- **Long Context Segmentation**: splitting retrieved passages and generated answers into concise, semantically coherent segments;
- **Semantic Evidence Aggregation**: embedding each segment into a shared vector space and constructing a segment graph whose edges encode pairwise semantic similarity; coherent clusters are identified via betweenness centrality to recover the logical evidence chains scattered across documents;
- **Consistency Scoring**: grouping related segments via graph traversal, applying a relevance module plus an NLI discriminator to each group, then aggregating scores into a global hallucination indicator.

Terminology Clarification. Our approach uses geometric clustering in embedding space based on semantic similarity, *not* formal logical operations ($\wedge$, $\vee$, $\Rightarrow$). It captures *semantic coherence* (the co-presence of related evidence across documents), which is a necessary prerequisite for multi-hop reasoning but is distinct from formal logical entailment.

Efficiency Positioning. This work targets practical deployment constraints. We demonstrate that graph-based evidence aggregation enables lightweight models (0.5B parameters) to achieve competitive accuracy (82.4%), placing our method at the Pareto-optimal frontier: $3.3\times$ faster than TrueTeacher (Gekhman et al., 2023) while using $22\times$ fewer parameters, and $6\times$ faster than GPT-4o with marginally *higher* accuracy.

To facilitate rigorous evaluation of multi-hop hallucination detection, we also contribute **HotPotQA-Derived**, derived from HotPotQA (Yang et al., 2018). Existing benchmarks either provide only a single related passage or merge multiple sources into one long context, obscuring cross-document links. Our dataset preserves separate retrieved documents, better reflecting real-world RAG pipelines.

Extensive experiments demonstrate that our framework, without any task-specific pretraining, surpasses larger LLM baselines on HotPotQA-Derived (82.4% accuracy) and RAGTruth (Niu et al., 2023) (72.2% F1), and achieves competitive performance on HaluBench and HaluEval.

2 Related Work

Fine-tuning on hallucination corpora. Lynx (Ravi et al., 2024), RAG-HAT (Song et al., 2024), and Osiris (Shan et al., 2025) improve hallucination detection by fine-tuning language models on annotated datasets. These methods assume all evidence can be loaded into a single context window, limiting their applicability in multi-document RAG settings where domain-specific annotated corpora may be unavailable.

NLI-based consistency checking. MiniCheck (Tang et al., 2024), AlignScore (Zha et al., 2023), and TrueTeacher (Gekhman et al., 2023) apply NLI for fact-checking. Provenance (Sankararaman et al., 2024) and RAGAS (Es

Aspect	GRAG	GRADA	Ours
Goal	Subgraph retrieval	Adversarial defense	Hallucination detection
Stage	Pre-generation	Pre-generation	Post-generation
Nodes	Full documents	Full documents	**Segments**
Algorithm	Ego-graph	PageRank	**Betweenness**

Table 1: Graph-based methods in RAG: our method is the first to use segment-level graphs for post-generation hallucination verification.

Symbol	Definition
$\ell(s)$	Token length of sentence s
$\mathsf{Chunk}(X;T)$	Chunking function segmenting text X with threshold T
$E(c_i)$	Embedding function mapping chunk c_i to $\mathbb{R}^d$
P_{ij}	Shortest path between nodes i and j
f_e	Frequency of edge e appearing in all-pairs shortest paths
$\tau(S)$	Total token count of cluster S
$R(D_k, A')$	Relevance scoring function
$\mathrm{NLI}(D_k, H)$	NLI entailment probability

Table 2: Notation and Symbol Definitions.

et al., 2025) repurpose NLI models to score factual consistency against individual retrieved documents. More recently, Grounded Context Retrieval (Gerner et al., 2025) highlights NLI's weaknesses on long, multi-hop contexts. However, none of these approaches explicitly model semantic coherence across multiple retrieved segments.

Graph-based methods in RAG. Recent work has explored graph-based representations within RAG pipelines; however, they address fundamentally different pipeline stages (Table 1). **GRAG** (Hu et al., 2024) retrieves K-hop ego-graphs from document collections to improve retrieval *before generation*. **GRADA** (Zheng et al., 2025) applies PageRank-style propagation among retrieved documents to detect adversarially injected documents *before generation*, addressing robustness rather than hallucination verification. Our method operates *after generation* with segment-level graphs (multiple nodes per document), uses betweenness centrality for evidence aggregation, and directly addresses multi-hop hallucination detection. These three approaches are complementary and can be chained as successive pipeline stages.

On the notion of "logic" in this work. The term *logic* in our title refers to the multi-hop inferential dependencies spanning multiple retrieved documents, i.e., the cross-document reasoning chains a hallucination detector must trace to judge whether an answer is grounded in the retrieved evidence. This is distinct from *formal logic operations* as studied in, e.g., visual classification (Tan et al., 2024), where logic denotes symbolic rules derived from propositional or predicate calculus. In our setting, hallucination arises from semantic inconsistencies across separate retrieval outputs rather than from violations of symbolically defined constraints; accordingly, our framework captures these dependencies through graph-based evidence aggregation rather than formal rule inference.

3 Methodology

3.1 Problem Statement

Lightweight hallucination detection for RAG must overcome three challenges. **(1) Limited NLI context window**: NLI models can only process finite text; when retrieved passages exceed this limit, essential facts may be omitted. **(2) Cross-passage dependencies**: standard NLI evaluates one passage at a time, ignoring evidence spanning multiple documents. **(3) Ambiguity in NLI scoring**: a low entailment score can arise from direct contradiction or mere irrelevance; retrieved passages unrelated to the generated answer can trigger false alarms.

Appendix A provides illustrative examples of these limitations.

3.2 Framework Overview

Our framework comprises three modules: Long Context Segmentation, Semantic Evidence Aggregation, and Consistency Scoring, each addressing one of the three challenges above. A sketch of the pipeline is in Figure 2.

3.3 Long Context Segmentation

Let $\ell(s)$ denote the token length of sentence s. Define thresholds T_a (answers) and T_d (documents). The chunking operation $\mathsf{Chunk}(X;T) = \{c_1, \ldots, c_m\}$ partitions text X into segments each satisfying $\sum \ell(s_\ell) \leq T$.

Factual Sentence Classifier $f(c)$. Let $f(c) \in \{0, 1\}$ be a rule-based binary classifier that filters non-factual sentences (questions, opinions, rhetorical statements) to focus graph construction on factual content. The classifier combines: (1) *regex patterns* detecting temporal facts, quantitative facts, and named entities; (2) *POS-tag signals* checking

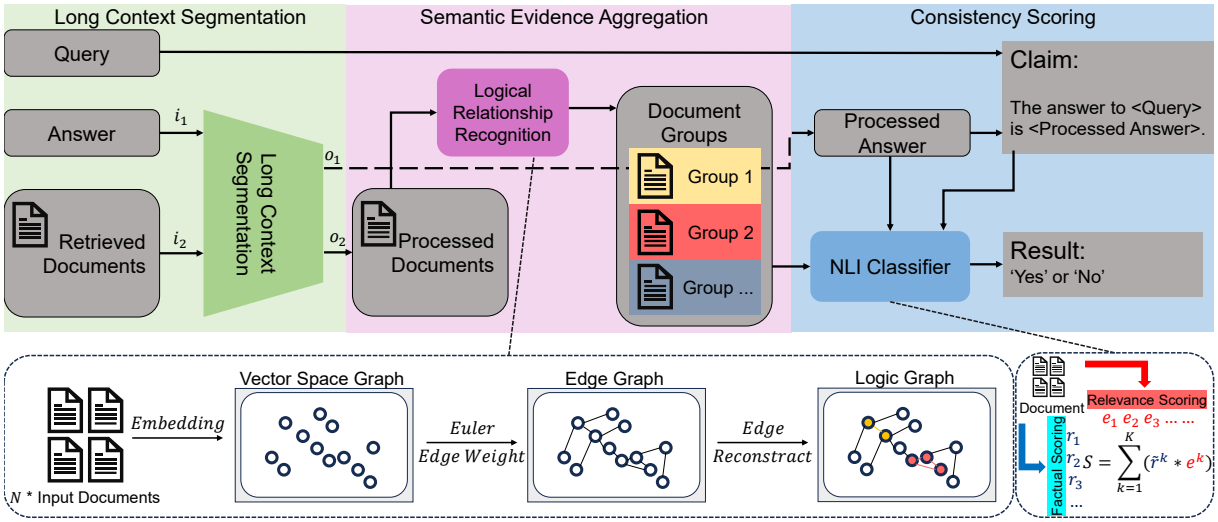

Figure 2: Our Hallucination Classifier Architecture, comprising three modules: Long Context Segmentation (green), Semantic Evidence Aggregation (pink), and Consistency Scoring (blue). Given a question-answer pair and retrieved documents, we (1) segment into factual chunks, (2) construct a semantic similarity graph and identify coherent clusters via betweenness centrality, and (3) verify each cluster against the answer using an NLI discriminator. This example shows how scattered evidence from Doc 1 and Doc 2 is successfully aggregated. Notation definitions in Table 2.

Metric	Value
Precision	87.3%
Recall	82.1%
F1-Score	84.6%
System accuracy (full)	82.4%
System accuracy (w/o filter)	78.2% (−4.2%)

Table 3: Performance of the factual sentence classifier $f(c)$ on 1,000 labeled sentences, and its contribution to end-to-end accuracy on HotPotQA-Derived *bridge-hard*. Full rule specification is in Appendix H.

for proper nouns (NNP/NNPS) or action verbs in indicative mood; and (3) a *rejection list* for interrogative openers, opinion markers, and imperative constructions. Table 3 reports its performance.

Answer chunking. Given answer A with $\ell(A) > T_a$, compute $C_a = \mathsf{Chunk}(A; T_a)$. Reassemble the factual content:

$$A' = \mathrm{Concat}\big(\{c \in C_a : f(c) = 1\}\big).$$

Document chunking. Given retrieved documents $\{d\}$, form $\mathcal{C} = \bigcup_d C_d$ where $C_d = \mathsf{Chunk}(d; T_d)$ if $\ell(d) > T_d$, else $C_d = \{d\}$.

3.4 Semantic Evidence Aggregation

Graph construction. For each chunk $c_i \in \mathcal{C}$, compute embedding $v_i = E(c_i) \in \mathbb{R}^d$. With $N = |\mathcal{C}|$, compute pairwise distances $d_{ij} = \|v_i - v_j\|_2$ and their mean μ. For $\alpha = 1.0$ (determined via held-out validation in $[0.5, 1.5]$), define the initial edge set $E^{(0)} = \{(i,j) \mid d_{ij} \le \alpha\mu\}$ with weights $w_{ij}^{(0)} = d_{ij}$.

Betweenness-centrality clustering. On graph $G^{(0)}$, compute all-pairs shortest paths P_{ij}. For each edge $e \in E^{(0)}$, set $f_e = |\{(i,j) : e \in P_{ij}\}|$ (betweenness frequency), and $w_e^{(1)} = f_e$. Sort edges descending by $w^{(1)}$; initialize $\mathcal{S} = \{\{i\}\}_{i=1}^N$. For each edge $e = (i,j)$ in order, let S_p, S_q contain i, j respectively. If $\tau(S_p) + \tau(S_q) \le T_t$, merge:

$$S_{\text{new}} = S_p \cup S_q, \qquad \mathcal{S} \leftarrow (\mathcal{S} \backslash \{S_p, S_q\}) \cup \{S_{\text{new}}\}.$$

Finally, for each cluster $S_k \in \mathcal{S}$:

$$D_k = \mathrm{Concat}\big(\{c_i : i \in S_k\}\big).$$

Relationship to Girvan–Newman. This module draws inspiration from the Girvan–Newman algorithm (Girvan and Newman, 2002; Słoczy'nski, 2020): edges with high betweenness serve as *bridges* between semantic communities. We invert the insight: rather than removing high-betweenness edges to detect communities, we *prioritize* them to merge clusters of related evidence. **Important distinction**: the graph is a *semantic similarity graph* (not a logical dependency graph); betweenness centrality identifies clusters of coherent segments, approximating evidence aggregation for multi-hop reasoning. Detailed theoretical justification is in Appendix B.

3.5 Consistency Scoring

NLI-Agnostic Modularity. Our framework is *training-free* and *NLI-agnostic*: any NLI model can serve as the discriminator without modifying the graph-based aggregation logic. This enables practitioners to select domain-appropriate NLI models (e.g., specialized medical/legal models), upgrade to stronger models without retraining, and balance accuracy vs. efficiency for deployment constraints. Experimental validation is provided in Section 4.5.

Scoring procedure. For each grouped document D_k and reassembled answer A', compute relevance and normalize:

$$r_k = R(D_k, A'), \qquad \tilde{r}_k = \frac{r_k}{\sum_{j=1}^{K} r_j}.$$

Construct hypothesis H = "The answer to 'Q' is: A'". The NLI entailment score is $e_k = \text{NLI}(D_k, H)$. With threshold T_s:

$$S = \sum_{k=1}^{K} \tilde{r}_k \cdot e_k.$$

If $S > T_s$, the output is classified as *non-hallucinated*; otherwise as *hallucinated*.

3.6 HotPotQA-Derived Benchmark Construction

We contribute **HotPotQA-Derived** to enable systematic multi-hop hallucination evaluation. Existing benchmarks either provide a single passage or merge multiple sources, unfairly favoring LLM-based classifiers.

Step 1: Source Data Selection. We build on HotPotQA (Yang et al., 2018), selecting all qualifying "bridge" questions that require explicit reasoning across exactly two documents, with unambiguous ground-truth answers. No cherry-picking is performed.

Step 2: Hallucinated Answer Generation. We generate three difficulty levels: *Bridge-Easy* (wrong entity, correct category), *Bridge-Medium* (partially correct, wrong detail), and *Bridge-Hard* (correct individual facts combined incorrectly), yielding 14,282 / 45,863 / 12,246 examples (72,391 total). Generation uses DeepSeek-V3 at temperature 0 with prompts provided in Appendix G.

Step 3: Two-Stage Quality Control. *Stage 1 (Automatic)*: contradiction detection and fluency filtering. *Stage 2 (Human)*: trained annotators verify plausibility, factual incorrectness, and cross-document dependency of flagged samples (1,173 out of 72,391 were flagged; all were validated with no factual errors found). Details in Appendix F.

During evaluation, each question is paired at random with its faithful or a hallucinated answer; classifiers predict the binary label and we report accuracy.

4 Experiments

4.1 Benchmark Datasets

We evaluate on four benchmarks. Detailed sizes are in Appendix D. **RAGTruth** (Niu et al., 2023): 17,790 LLM responses from 2,965 query instances across QA, Data-to-Text, and Summarization tasks. **HaluBench** (Ravi et al., 2024): 13,867 samples across diverse domains, including hard-to-detect hallucinations. **HaluEval** (Li et al., 2023): 35,000 samples across QA, dialogue, and summarization tasks with manual and automatic subsets. **HotPotQA-Derived**: our new benchmark described in Section 3.6.

4.2 Experimental Settings

Model Components. Our framework integrates three off-the-shelf models (no task-specific training): **HHEM-2.1-Open** (Bao et al., 2024) (NLI discriminator, 60 M parameters), **mxbai-rerank-base-v2** (Lee et al., 2025) (relevance scorer, 480 M parameters), and **all-MiniLM-L6-v2** (Reimers and Gurevych, 2019) (sentence encoder, 22 M parameters); plus parameter-free graph operations (NetworkX). Total: **≈0.56 B parameters** (Config A). Replacing HHEM with MiniCheck-FT5 (Tang et al., 2024) (∼1 B) yields Config B (≈1.5 B total).

Hyperparameters. We tune on held-out validation splits and use $C = 256$, $T_a = T_d = 512$, $T_t = 1{,}024$, $T_s = 0.4$ across all benchmarks unless stated otherwise.

Hardware. Efficiency experiments are conducted on RTX 4090; main results were originally measured on RTX 4060 (8 GB) to represent a resource-constrained scenario.

4.3 Main Results

RAGTruth. On RAGTruth (Table 4), Config A surpasses all non-pretrained baselines, attaining best F1 in QA (85.8%) and Summarization (73.7%). Performance on Data-to-Text is lower; this subset uses structured tables rather than free-form text, where our semantic similarity graph is less effective; analysis in Appendix D.

Pretrained	Method	Params	QA			Data-to-Text			Summarization			Overall		
			P	R	F1	P	R	F1	P	R	F1	P	R	F1
	LLM Methods													
✓	Finetuned Llama-2-13B	13B	61.6	76.3	68.2	85.4	91.0	88.1	64.0	54.9	59.1	76.9	80.7	78.7
✓	RAG-HAT	–	76.5	73.1	74.8	92.9	90.3	91.6	77.7	59.8	67.6	87.3	80.8	83.9
✓	Luna	–	37.8	80.0	51.3	64.9	91.2	75.9	40.0	76.5	52.5	52.7	86.1	65.4
✗	$\text{Prompt}_{\text{gpt-3.5}}$	∼175B	18.8	84.4	30.8	65.1	95.5	77.4	23.4	89.2	37.1	37.1	92.3	52.9
✗	$\text{Prompt}_{\text{gpt-4-turbo}}$	∼2000B	33.2	90.6	45.6	64.3	100.0	78.3†	31.5	97.6	47.6	46.9	97.9	63.4
✗	$\text{SelfCheckGPT}_{\text{gpt-3.5}}$	∼175B	35.0	58.0	43.7	68.2	82.8	74.8	31.1	56.5	40.1	49.7	71.9	58.8
✗	$\text{LMvLM}_{\text{gpt-4-turbo}}$	∼2000B	18.7	76.9	30.1	68.0	76.7	72.1	23.3	81.9	36.2	36.2	77.8	49.4
	NLI Methods													
✗	Provenance	∼0.1B	17.8	100.0	30.2	64.3	100.0	78.3†	23.8	81.4	36.8	36.2	96.0	52.6
✗	RAGAS Faithfulness	∼0.1B	18.1	63.1	28.1	66.0	89.7	76.0	20.4	66.5	31.2	27.5	73.4	40.0
✗	HHEM-2.1-Open	0.06B	20.4	73.1	31.9	64.6	99.5	78.3†	22.7	98.5	36.9	38.0	94.8	54.3
✗	MiniCheck-FT5	∼1B	25.7	38.1	30.7	64.7	84.5	73.3	47.2	45.1	46.1	54.0	68.1	60.3
✗	TrueTeacher	11B	31.5	32.5	32.0	78.4	55.3	64.9	49.3	33.8	40.1	61.9	46.8	53.3
✗	AlignScore-Base	0.2B	42.4	40.0	41.2	79.7	28.5	42.0	68.4	6.4	11.7	64.2	25.7	36.7
✗	AlignScore-Large	0.4B	39.1	33.8	36.2	71.6	54.1	61.6	52.5	20.6	29.6	62.4	43.8	51.2
✗	**Ours (Config A)**	**0.5B**	89.8	82.1	85.8*	39.7	42.4	41.0	80.7	67.2	73.7*	75.8	69.0	72.2*

Table 4: Response-level hallucination detection on RAGTruth (Niu et al., 2023). ∗ best among non-pretrained methods; underline best overall; † tied. "–" = parameter count not publicly disclosed.

Group	Method	Params	Easy	Med	Hard	Avg
	Small Models (<1B Parameters)					
NLI	HHEM-2.1-Open	0.06B	55.1	52.4	52.5	52.9
NLI	AlignScore-Base	0.2B	51.1	49.8	50.0	50.1
NLI	AlignScore-Large	0.4B	52.4	50.7	49.9	50.9
NLI	RAGAS Faithfulness	–	55.1	49.9	50.2	51.0
NLI	Provenance	–	54.5	52.2	52.4	52.7
LLM	Qwen3-0.6B	0.6B	48.9	48.8	49.1	48.9
LLM	Qwen3-0.6B-reason	0.6B	69.6	67.0	66.1	67.4
	Ours (Config A)	**0.5B**	**80.6**	**82.9**	**82.7**	**82.4**
	Large Models (≥1B Parameters)					
NLI	MiniCheck-FT5	∼1B	84.3	80.1	78.3	80.6
NLI	TrueTeacher	11B	86.5	85.2	81.1	84.8
LLM	GPT-3.5-Turbo	∼175B	42.3	44.1	43.9	43.7
LLM	GPT-4o	∼2000B	86.0	80.1	79.8	81.2
	Ours (Config B)	**1.5B**	**86.9**	**85.4**	**85.0**	**85.6**

Table 5: Accuracy (%) on HotPotQA-Derived grouped by parameter size for fair comparison. Config A (≈0.5B) outperforms all sub-1B baselines by >30%; Config B (≈1.5B) surpasses 11B TrueTeacher. Easy/Med/Hard correspond to `bridge-easy`/`bridge-medium`/`bridge-hard`; Avg is the full-dataset average.

HotPotQA-Derived. Table 5 groups models by parameter count. Within the <1B range, Config A (82.4%) outperforms all comparable-sized baselines by >30 percentage points. Accuracy remains stable from `bridge-easy` through `bridge-hard`, demonstrating that Semantic Evidence Aggregation effectively models multi-segment dependencies. Config B (MiniCheck discriminator, ≈1.5B) achieves 85.6%, surpassing the 11B TrueTeacher baseline. Figure 3 visualizes the Pareto landscape.

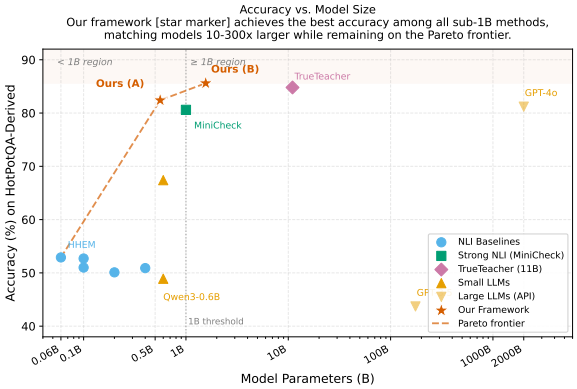

Figure 3: Accuracy vs. parameter count on HotPotQA-Derived. Our approach (starred marker) achieves the best accuracy among all sub-1B methods, approaching performance of models 10–300× larger. The Pareto frontier (dashed line) is occupied only by our method in the <1B range.

HaluEval & HaluBench. Our method consistently outperforms NLI-based baselines across HaluBench (70.1% Acc, Config A; 78.8%, Config B) and HaluEval (58.1%, Config A; 58.0%, Config B). Detailed tables are in Appendix D.

4.4 Comprehensive Efficiency Analysis

Table 6 and Figure 4 analyze the efficiency–performance trade-off. Key findings:

Category	Method	ms	q/s	GB
NLI-Based	HHEM-2.1-Open	25±2	400	0.5
	AlignScore-Large	42±2	238	1.7
	MiniCheck-FT5	165±8	61	4.2
LLM-Based	Qwen3-0.6B	120±6	83	2.8
	GPT-4o (API)	520±67	19*	N/A
	TrueTeacher 11B	280±8	36	16.2
Ours	Config A (0.5B)	**85±4**	**118**	**2.1**
Ours	Config B (1.5B)	165±8	61	4.8

Table 6: Inference efficiency on RTX 4090 (latency / throughput / memory). *API throughput is rate-limit-bounded. Full comparison across all baselines including component-level breakdown is in Appendix C.

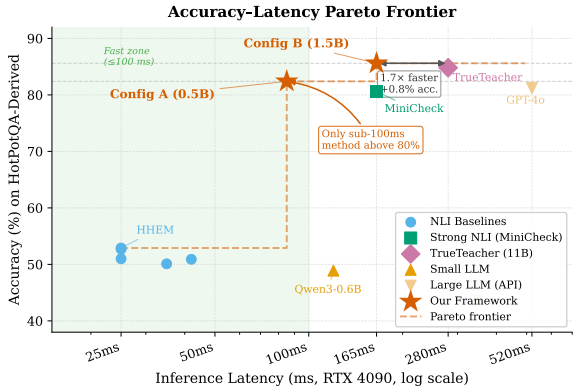

Figure 4: Accuracy vs. latency (log scale) on HotPotQA-Derived. Methods on the Pareto frontier (dashed curve) cannot be improved on one axis without sacrificing the other. Both configurations of our method occupy the Pareto frontier: Config A is the fastest among all methods achieving >80% accuracy; Config B surpasses TrueTeacher at 1.7× lower latency.

- **vs. fast NLI-only methods**: Standalone NLI is 3–5× faster but achieves near-random accuracy (50–55%) on multi-hop tasks, making them impractical despite low latency.
- **vs. accurate methods**: Config A (82.4%, 118 q/s) is 3.3× faster than TrueTeacher (84.8%, 36 q/s) with only 2.4% lower accuracy, and 6.1× faster than GPT-4o with 1.2% *higher* accuracy.
- **Memory**: At 2.1 GB, Config A is deployable on consumer GPUs (vs. 16–18.5 GB for TrueTeacher or Llama3.1-8B).

Component-level analysis. Profiling (detailed in Appendix C) shows NLI inference dominates runtime (63.7%), while Semantic Evidence Aggregation adds only ∼18.5 ms (21.8% of total). The return on investment for graph clustering is 6.2% accuracy gain / 18 ms = 0.34, comparable to the full pipeline ROI of 14.4%/43 ms = 0.33, confirming the module's efficiency.

NLI Model	Params	Alone	+Ours	Δ
HHEM-2.1-Open (default)	0.06B	52.5%	82.4%	+29.9%
nli-roberta-base	0.1B	50.2%	66.9%	+16.7%
DeBERTa-v3-base	0.3B	54.7%	78.3%	+23.6%
ModernBERT-large-nli	0.4B	52.0%	75.2%	+23.2%
AlignScore-Large	0.4B	49.9%	70.6%	+20.7%
MiniCheck-FT5	∼1B	78.3%	85.0%	+6.7%
TrueTeacher	11B	81.1%	80.2%	−0.9%

Table 7: Framework effectiveness across seven NLI discriminators on HotPotQA-Derived Hard. Six of seven models improve (+6.7%–+29.9%); TrueTeacher (11B) shows a marginal decline (−0.9%) as its larger capacity already handles long-context reasoning without additional graph support.

4.5 Framework Effectiveness Across NLI Models

Table 7 and Figure 6(a) demonstrate that our framework is broadly NLI-agnostic: it improves six of seven tested discriminators. Key findings: (1) Standalone NLI achieves near-random accuracy (50–55%) on multi-hop tasks, motivating the need for evidence aggregation. (2) Our graph module improves most discriminators by +6.7% to +29.9%, adding only ∼18 ms overhead. (3) Lighter discriminators benefit more: HHEM gains +29.9% because its small capacity limits long-range reasoning, which the graph module compensates for. (4) TrueTeacher (11B) shows a marginal decline (−0.9%), consistent with its larger parameter count already providing sufficient long-context perception. (5) Combining with MiniCheck (Figure 6(b)) achieves 85.6%, surpassing TrueTeacher while using 7× fewer parameters. Removing the graph module (ablation, Table 8) causes a 19.6% accuracy drop, confirming it as the primary performance driver.

4.6 Parameter Sensitivity Analysis

Figure 5 shows robustness across three key hyperparameters. **Chunk size** C: accuracy remains above 75% for $C \geq 128$, but drops >5 points at $C = 64$ (over-segmentation fragments evidence). **Segmentation thresholds**: stable for $384 \leq T \leq 1024$; falls below 75% at $T = 256$, confirming over-segmentation as a more critical failure mode than handling longer contexts. **Decision threshold** T_s: broad plateau around 80% for $0.3 \leq T_s \leq 0.7$,

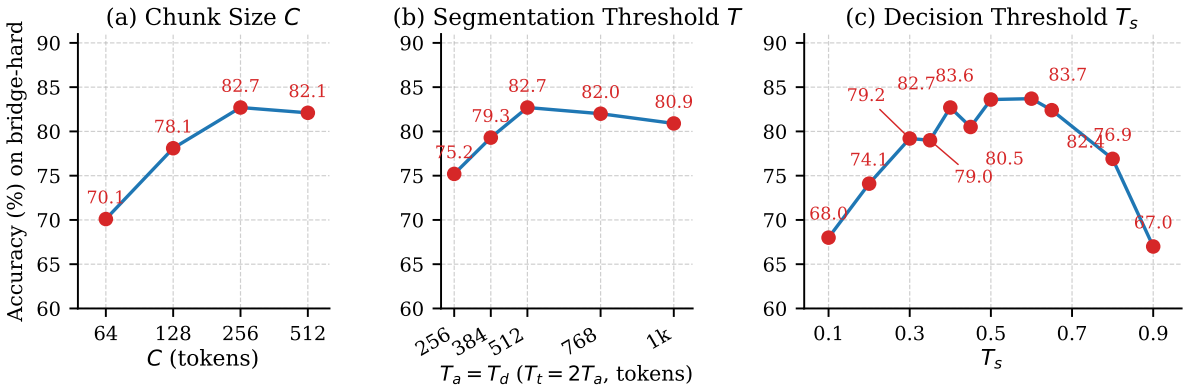

Figure 5: Accuracy on HotPotQA-Derived *bridge-hard* with (a) Chunk Size C, (b) Segmentation Thresholds T, and (c) Decision Threshold T_s. All hyperparameters exhibit broad stable plateaus, confirming robustness to configuration choice.

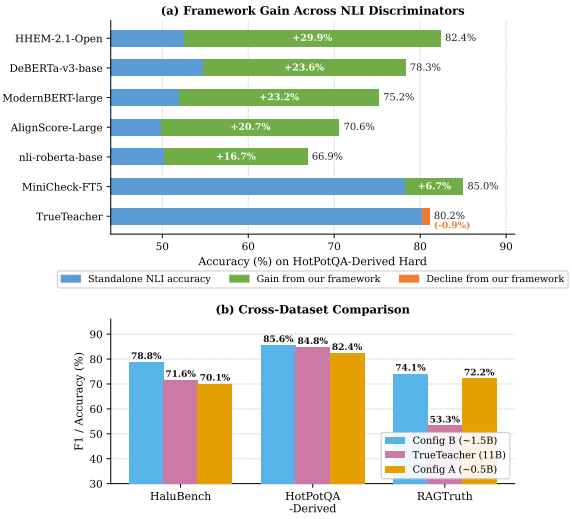

Figure 6: **(a)** Framework accuracy vs. standalone NLI accuracy for seven discriminators on HotPotQA-Derived Hard. Six of seven models improve; TrueTeacher marginally declines (-0.9%) as its 11B capacity already handles long-context reasoning. **(b)** Cross-dataset comparison of Config A ($\sim$0.5B) and Config B ($\sim$1.5B) against TrueTeacher (11B) across three benchmarks.

Configuration	Acc (%)	Lat. (ms)
Full (A+B+C)	82.7*	85
w/o A (LC)	78.2 (-4.5)	73
w/o B (SEA)	63.1 (-19.6)	67
w/o C (CS)	59.4 (-23.3)	31
w/o A+B	55.0 (-27.7)	55
w/o B+C	51.9 (-30.8)	19
w/o A+C	52.1 (-30.6)	60

Table 8: Ablation on HotPotQA-Derived *bridge-hard*. A = Long Context Segmentation (LC), B = Semantic Evidence Aggregation (SEA), C = Consistency Scoring (CS). Removing B alone costs 19.6% for only 18 ms saved, confirming SEA as the core contribution. *Best result.

confirming that combined relevance–entailment scoring is highly discriminative. Chunk size vs. latency trade-offs are detailed in Appendix C.

4.7 Ablation Study

Table 8 quantifies individual module contributions. Removing Semantic Evidence Aggregation (B) causes the largest single-module drop (-19.6%) while saving only 18 ms (21% of latency), the worst accuracy–efficiency trade-off among all ablations. All dual-module removals drop accuracy below 55% (near random), confirming that the three modules are complementary and mutually necessary.

5 Conclusion

We present a lightweight, NLI-agnostic hallucination detection framework for RAG systems built around graph-based Semantic Evidence Aggregation. The framework operates without task-specific pretraining and provides two deployment configurations: Config A (0.5B, 82.4%, 85 ms) achieves state-of-the-art accuracy among sub-1B methods and is Pareto-optimal in the efficiency–performance space; Config B (1.5B, 85.6%, 165 ms) surpasses the 11B TrueTeacher baseline at $1.7\times$ lower latency. Across seven NLI discriminators, the graph module provides consistent gains of $+6.7\%$ to $+29.9\%$, validating that evidence aggregation, not the specific NLI model, is the primary performance driver.

The plug-and-play design decouples graph-based evidence aggregation from NLI scoring, so

any off-the-shelf discriminator can benefit without system retraining. Combined with sub-200 ms per-sample latency and a 2.1 GB memory footprint for Config A, the framework is practical for production deployment on consumer hardware. We also contribute HotPotQA-Derived, a three-tier multi-hop hallucination benchmark (Easy / Medium / Hard) that enables difficulty-stratified evaluation beyond what single-hop benchmarks support, and we hope it will serve as a standard testbed for future research on multi-hop hallucination detection in RAG systems.

Limitations

Semantic vs. Logical Gap. Our Semantic Evidence Aggregation module captures semantic coherence in embedding space but does not perform formal logical operations. For queries requiring multi-step conditional inference, the NLI discriminator must still handle the conditional structure, which may exceed the capacity of small-scale NLI models even when evidence is correctly aggregated. Future work could integrate symbolic reasoning modules as a downstream verification step over our aggregated evidence clusters.

Coarse-Grained Semantic Relations. The module merges chunks based on aggregate graph connectivity, treating all semantic relations uniformly. It does not differentiate between relation types (causation, temporality, coordination). Concatenated segments may therefore not preserve causally coherent ordering.

Embedding Model Robustness. Our primary implementation uses `all-MiniLM-L6-v2` as the sentence encoder. While theoretical arguments and the literature suggest that semantic graph structure is preserved across well-trained embedders (Tavares et al., 2024), a comprehensive empirical study of embedding model variation remains an open item for future work.

Data-to-Text Performance. Our method shows weaker performance on the RAGTruth Data-to-Text subset, likely because this subset is generated from structured tables where semantic proximity in embedding space is less effective at capturing key-value relationships.

Acknowledgments

The work was partially supported by the following: National Natural Science Foundation of China under No. 92370119, and 62376113.

References

Forrest Bao, Miaoran Li, Rogger Luo, and Ofer Mendelevitch. 2024. HHEM-2.1-Open.

Shahul Es, Jithin James, Luis Espinosa-Anke, and Steven Schockaert. 2025. Ragas: Automated evaluation of retrieval augmented generation. *Preprint*, arXiv:2309.15217.

Zorik Gekhman, Jonathan Herzig, Roee Aharoni, Chen Elkind, and Idan Szpektor. 2023. Trueteacher: Learning factual consistency evaluation with large language models. *Preprint*, arXiv:2305.11171.

Assaf Gerner, Netta Madvil, Nadav Barak, Alex Zaikman, Jonatan Liberman, Liron Hamra, Rotem Brazilay, Shay Tsadok, Yaron Friedman, Neal Harow, Noam Bresler, Shir Chorev, and Philip Tannor. 2025. Grounded in context: Retrieval-based method for hallucination detection. *Preprint*, arXiv:2504.15771.

Michelle Girvan and Mark EJ Newman. 2002. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826.

Yuntong Hu, Zhihan Lei, Zheng Zhang, Bo Pan, Chen Ling, and Liang Zhao. 2024. GRAG: Graph retrieval-augmented generation. *arXiv preprint arXiv:2405.16506*.

Sean Lee, Rui Huang, Aamir Shakir, and Julius Lipp. 2025. Baked-in brilliance: Reranking meets rl with mxbai-rerank-v2.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. *CoRR*, abs/2005.11401.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6449–6464, Singapore. Association for Computational Linguistics.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, Kashun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2023. Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. *arXiv preprint arXiv:2401.00396*.

Selvan Sunitha Ravi, Bartosz Mielczarek, Anand Kannappan, Douwe Kiela, and Rebecca Qian. 2024. Lynx: An open source hallucination evaluation model. *Preprint*, arXiv:2407.08488.

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*. Association for Computational Linguistics.

Hithesh Sankararaman, Mohammed Nasheed Yasin, Tanner Sorensen, Alessandro Di Bari, and Andreas Stolcke. 2024. Provenance: A light-weight fact-checker for retrieval augmented LLM generation output. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1305–1313, Miami, Florida, US. Association for Computational Linguistics.

Alex Shan, John Bauer, and Christopher D. Manning. 2025. Osiris: A lightweight open-source hallucination detection system. *Preprint*, arXiv:2505.04844.

Adam Słoczy'nski. 2020. An overview of algorithms for community detection in networks.

Juntong Song, Xingguang Wang, Juno Zhu, Yuanhao Wu, Xuxin Cheng, Randy Zhong, and Cheng Niu. 2024. RAG-HAT: A hallucination-aware tuning pipeline for LLM in retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1548–1558, Miami, Florida, US. Association for Computational Linguistics.

Zhaorui Tan, Xi Yang, Qiufeng Wang, Anh Nguyen, and Kaizhu Huang. 2024. Interpret your decision: Logical reasoning regularization for generalization in visual classification. In *Advances in Neural Information Processing Systems*, volume 37.

Liyan Tang, Philippe Laban, and Greg Durrett. 2024. Minicheck: Efficient fact-checking of llms on grounding documents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Tiago F. Tavares, Fabio Ayres, and Paris Smaragdis. 2024. Measuring similarity between embedding spaces using induced neighborhood graphs. *Preprint*, arXiv:2411.08687.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. Alignscore: Evaluating factual consistency with a unified alignment function. *Preprint*, arXiv:2305.16739.

Jingjie Zheng, Aryo Pradipta Gema, Giwon Hong, Xuanli He, Pasquale Minervini, Youcheng Sun, and Qiongkai Xu. 2025. GRADA: Graph-based reranking against adversarial documents attack. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 22244–22266.

A Challenges of NLI Models

Tables 9 and 10 illustrate the limitations of existing NLI models in factual scoring. When parts of a logical chain are treated as independent texts, their factuality scores are much lower than for the complete text (Table 9). NLI classifiers also struggle to distinguish between contradiction and irrelevance (Table 10).

Premise	Hypothesis	Score
CityA is a capital city.	Capital of CountryA is CityA.	0.030
CityA is a city of CountryA.	Capital of CountryA is CityA.	0.117
CityA is a capital city. CityA is a city of CountryA.	Capital of CountryA is CityA.	0.632

Table 9: Combining partial premises dramatically improves entailment scores.

Premise	Hypothesis	Score
I am a dog.	I am a cat.	0.023
I am a dog.	You are a cat.	0.022

Table 10: Contradiction and irrelevance produce similarly low NLI scores.

B Rationale of Semantic Evidence Aggregation

The module is effective for three reasons.

Community-bridge intuition. The Girvan–Newman algorithm (Girvan and Newman, 2002) detects communities by removing high-betweenness edges. We invert this insight: edges with high $f_e = |\{(i,j) : e \in P_{ij}\}|$ serve as bridges between semantic communities; prioritizing them merges clusters of related evidence.

Filtering irrelevant chunks. An edge $(i,j) \notin E^{(0)}$ whenever $d_{ij} > \alpha\mu$, preventing merging of unrelated segments.

Avoiding redundant merges. Because Euclidean distance satisfies the triangle inequality $d_{ij} \leq d_{ik} + d_{kj}$, intra-community edges have low betweenness and are deprioritized, selecting only true inter-community bridges.

Illustrative example. For the query "What football position did the manager of Tianjin Quanjian previously hold?" with answer "Captain", the baseline NLI approach misclassifies the answer (maximum consistency score 0.115 $<$ threshold 0.4)

because no single passage alone identifies Fabio Cannavaro as both manager and former captain. After Semantic Evidence Aggregation merges the two relevant passages, the consistency score rises to 0.842, yielding the correct classification. Full retrieved document tables are in the supplementary materials.

C Comprehensive Efficiency Data

Table 11 reports full per-method latency, throughput, and GPU memory for all baselines and our configurations on an RTX 4090. Table 12 breaks down pipeline latency by module for Config A. Table 13 shows how chunk size C jointly affects accuracy and latency, complementing the analysis in Section 4.4.

Category	Method	ms	q/s	GB
NLI	HHEM-2.1-Open	25±2	400	0.5
	nli-roberta-base	28±2	357	0.6
	DeBERTa-v3-base	38±2	263	1.5
	ModernBERT-large	40±2	250	1.6
	AlignScore-Large	42±2	238	1.7
	MiniCheck-FT5	165±8	61	4.2
LLM	Qwen3-0.6B	120±6	83	2.8
	Llama3.1-8B	450±12	22	18.5
	GPT-3.5-Turbo (API)	350±45	29*	N/A
	GPT-4o (API)	520±67	19*	N/A
Other	TrueTeacher 11B	280±8	36	16.2
Ours	Config A (0.5B)	**85±4**	**118**	**2.1**
Ours	Config B (1.5B)	165±8	61	4.8

Table 11: Full latency comparison on RTX 4090. *API throughput limited by rate limits.

Component	ms	Std	%	Driver
LC Segmentation	12.3	0.8	14.5	POS tagging
Sent. split	3.1	0.2	3.6	spaCy
Rule filter	6.8	0.5	8.0	Regex+POS
Chunk assem.	2.4	0.3	2.8	Text ops
SEA Module	18.5	1.2	21.8	Graph constr.
Embedding	8.2	0.6	9.6	SBERT
Graph constr.	4.7	0.4	5.5	Similarity
Betweenness	5.6	0.7	6.6	NetworkX
Consistency Scoring	54.2	2.8	63.7	NLI infer.
NLI infer.	48.3	2.5	56.8	Cross-enc.
Score aggr.	5.9	0.6	6.9	Wt. sum
Total	**85.0**	4.1	100	

Table 12: Component-level timing analysis (RTX 4090, Config A, $C = 256$). Our graph contribution (21.8% of time) provides $+14\%$ accuracy over standalone NLI at only 43 ms additional overhead.

C	Chunks	Nodes	Acc (%)	Seg (ms)	Graph (ms)	Total (ms)
256	42	38	82.7%	12	17	77
384	29	26	82.4%	10	14	62
512	23	20	81.8%	9	12	53
768	16	14	80.9%	7	10	41
1024	12	11	79.2%	7	8	33

Table 13: Chunk size impact on accuracy and latency (RTX 4090, Config A). For latency-critical applications, $C = 512$ offers -0.9% accuracy at $+31\%$ speedup.

Test Set	Subset	# Samples
HaluBench	—	13,867
HaluEval	Manual	4,507
	Auto (QA)	10,000
	Auto (Dialogue)	10,000
	Auto (Summ.)	10,000
RAGTruth Test	QA	989
	Data-to-Text	1,033
	Summarization	943
HotPotQA-Derived	bridge-easy	14,282
	bridge-medium	45,863
	bridge-hard	12,246
Total	—	123,730

Table 14: Sizes of all test sets used in experiments.

Method	Params	Easy	Med.	Hard
NLI Methods				
HHEM-2.1-Open	0.06B	55.1	52.4	52.5
AlignScore-Base	0.2B	51.1	49.8	50.0
AlignScore-Large	0.4B	52.4	50.7	49.9
MiniCheck-FT5	∼1B	84.3	80.1	78.3
TrueTeacher	11B	86.5	85.2	81.1
LLM Methods				
Qwen3-0.6B	0.6B	48.9	48.8	49.1
Qwen3-0.6B-reason	0.6B	69.6	67.0	66.1
GPT-3.5-Turbo	∼175B	42.3	44.1	43.9
GPT-4o	∼2000B	86.0	80.1	79.8
Ours (Config A)	0.5B	80.6	82.9	82.7
Ours (Config B)	1.5B	86.9	85.4	85.0

Table 15: Full per-level accuracy (%) on HotPotQA-Derived.

D Dataset Details and Additional Results

RAGTruth Data-to-Text Analysis. Our method achieves mediocre performance on the Data-to-Text subset (41.0% F1 vs. 73.7% for Summarization). RAGTruth Data-to-Text responses are generated from structured tables; semantic similarity in embedding space is less effective at capturing key-value relationships that lack natural language proximity cues. Multi-hop reasoning is less relevant in this setting where each piece of information is self-contained.

Model	Acc (%)
LLM Methods	
LYNX (70B)	88.4
GPT-4o	87.9
Llama-3-70B-Instruct	87.0
GPT-4-Turbo	86.0
LYNX (8B)	85.7
Claude-3-Sonnet	84.5
Llama-3-8B-Instruct	83.1
Mistral-7B-Instruct	78.3
Claude-3-Haiku	68.9
GPT-3.5-Turbo	62.2
NLI Methods	
TrueTeacher (11B)	71.6
MiniCheck-FT5	69.1
Provenance	65.6
AlignScore-Large	61.2
HHEM-2.1-Open	60.5
AlignScore-Base	58.4
RAGAS Faithfulness	56.9
Ours – Config A (0.5B)	**70.1**
Ours – Config B (1.5B)	**78.8**

Table 16: Overall accuracy (%) on HaluBench (Ravi et al., 2024). Config B surpasses all NLI-based methods and is competitive with 7B LLMs.

Model	Acc (%)
LLM Methods	
Davinci002/003	80.4
ChatGPT	79.4
Claude 2	75.0
GPT-3	72.7
NLI Methods	
TrueTeacher (11B)	63.9
Provenance	56.7
AlignScore-Large	51.7
AlignScore-Base	50.7
MiniCheck-FT5	51.6
HHEM-2.1-Open	50.1
RAGAS Faithfulness	53.6
Ours – Config A (0.5B)	**58.1**
Ours – Config B (1.5B)	**58.0**

Table 17: Accuracy (%) on HaluEval (Li et al., 2023). The marginal difference between Config A and Config B is consistent with MiniCheck's documented underperformance on synthetic HaluEval errors.

E RAGAS Faithfulness: Categorization Note

The RAGAS Faithfulness baseline uses the `FaithfulnesswithHHEM` variant. Although it invokes an LLM to decompose claims into substatements, the final faithfulness judgment relies on an NLI model (HHEM) scoring each claim against the retrieved context, making it conceptually closer to an NLI-based approach. Furthermore, the faithfulness accuracy is not sensitive to which LLM performs the decomposition step. For these reasons, RAGAS Faithfulness is categorized as an NLI method in all result tables and excluded from parameter count discussions.

F HotPotQA-Derived: Quality Control Details

Initial Quality Control. During data generation, we conducted stratified sampling at approximately 1:500. The concise answer styles of HotPotQA make it nearly impossible for generated hallucinated answers to be factually incorrect when following this style.

Comprehensive Quality Validation. LLMs flagged 1,173 potentially problematic samples out of 72,391. Manual secondary verification on all flagged samples confirmed no factual generation errors.

G Prompts for Data Generation

All generation and screening phases use DeepSeek-V3 at temperature 0.

Generation System Prompt:

```
"You are an expert in generating subtly
↪  hallucinated answers."
"Your task is to create responses that appear
↪  credible at first
 glance, but contain verifiable factual errors
 ↪  when cross-checked
 with the provided golden answer and context."
```

Generation User Prompt:

```
"Question: <question>"
"Golden Answer: <answer>"
"Context: <context>"
"Generate a plausible but factually incorrect
↪  answer that:
 1. Maintains grammatical correctness
 2. Contains subtle factual inconsistencies
 3. Presents logical reasoning flaws
 4. Includes inaccurate numerical/data references
 5. If Golden Answer answers briefly with a noun
 ↪  or phrase,
    you should do the same"
```

Screening System Prompt:

```
"You are a cautious validator for hallucinated
↪  answers."
"Flag answers where the requested hallucination
↪  is:
 1. Not clearly detectable (errors are ambiguous
 ↪  or borderline)
 2. Potentially non-existent (accidentally
 ↪  correct/plausible)
 3. Insufficiently severe (fails to meet the
 ↪  defined criteria)"
"Priority: Minimize missed cases, even at the
↪  cost of
 over-flagging."
```

Evaluation Prompt (for LLM baselines):

```
System: "You are an expert in verifying
↪  hallucination.
        Please judge if the hallucination exists
        ↪  in the
        answer of query given contexts.
        If hallucination exists, print 'Yes'.
        ↪  Else, 'No'."
User:   "Query: <query>
        Answer: <answer>
        Context: <context>"
```

H Rule-Based Factual Sentence Classifier

The rule-based classifier $f(c) \in \{0, 1\}$ identifies whether a text chunk c contains a verifiable factual claim worthy of NLI scoring.

Filter Conditions (applied in order). A chunk is classified as *factual* ($f(c) = 1$) if it satisfies all:

1. **Length**: token count ≥ 5.
2. **Verb presence**: at least one finite verb detected by POS tagging.
3. **Named entity or numeric**: contains at least one named entity (PERSON, ORG, GPE, DATE, CARDINAL, etc.) or numeric token.
4. **Declarative structure**: sentence-final token is not a question mark or imperative-only verb form.
5. **Rejection list**: chunk does not consist solely of transitional phrases (e.g., "Furthermore," "However,") without a substantive claim.

Performance. On 1,000 labeled sentences: P = 87.3%, R = 82.1%, F1 = 84.6%. Common false positives include figurative language; common false negatives include colloquial factual statements.

Sensitivity Analysis. Relaxing filter condition (3) (removing the entity requirement) increases recall by 2.1% but decreases precision by 5.8%, resulting in a net F1 loss of 1.7%. The default configuration represents the best precision–recall trade-off on the validation set.

I Convex Hull Volume Analysis

We designed a Monte Carlo algorithm to estimate the convex hull volume of segment embeddings and investigated whether it could serve as a quality metric for embedding models. Intuitively, a larger convex hull volume indicates greater feature diversity. Preliminary experiments suggest a positive correlation between convex hull volume and hallucination detection accuracy for models of identical dimensionality (Model A: relative volume 1.00, accuracy 82.4%; Model B: relative volume 1.14, accuracy 83.1%). However, computing high-dimensional convex hulls is computationally expensive, and this analysis requires broader validation across datasets and model families. We include it here for reference only; no conclusions are drawn in the main text.