UAHV: UNCERTAINTY-AWARE HIERARCHICAL VERIFICATION FOR HALLUCINATION DETECTION IN LARGE LANGUAGE MODELS

ABSTRACT

To mitigate hallucinations in large language models (LLMs), recent methods have increasingly adopted fine-grained decomposition or repeated reasoning to improve factual verification. However, they often apply predefined verification procedures across samples, overlooking the fact that factual conflicts can differ in scope, evidence requirements, and decision uncertainty. This mismatch may lead to either missed conflicts or erroneous factual judgments. To address this limitation, we propose Uncertainty-Aware Hierarchical Verification (UAHV), an adaptive framework that adjusts verification according to sample-specific factual states. UAHV introduces two complementary mechanisms: Hierarchical Factual Verification (HFV) progressively routes samples from global semantic screening to local verification and selectively invokes arbitration for uncertain cases, while Type-Aware Asymmetric Verification (TAV) applies different evidence-matching strategies to evidence-sensitive and semantically flexible facts. Experiments on three datasets and four LLM backbones show that UAHV achieves the best F1 score in 11 of 12 settings, with an average F1 of 76.12%, outperforming the strongest baseline by 6.27 percentage points.

1 INTRODUCTION

LLMs have shown exceptional performance in natural language understanding and generation (Aracena et al., 2024; Fan et al., 2025). They integrate multi-source information to produce content responses (Kong et al., 2025). However, their generated content may suffer from hallucinations (Bang et al., 2025; Xiong et al., 2023; Liu et al., 2025a), which can lead to severe consequences in high-stakes domains such as law and finance (Lee et al., 2025). This risk makes hallucination detection crucial, as it serves as an initial review mechanism for content generated by large language models and helps mitigate potential harm (Li et al., 2026a). A central challenge in such detection is how to determine the appropriate granularity for factual verification: whether a response should be assessed holistically or decomposed into smaller factual units for fine-grained checking (Gunjal & Durrett, 2024).

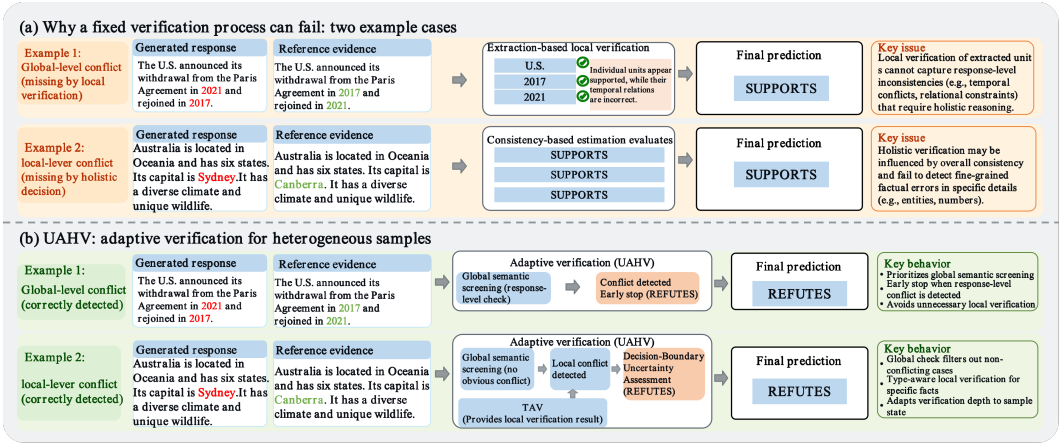


Figure 1: Motivation of UAHV: From Predefined Verification to Adaptive Hierarchical Verification.

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.

Consistency-based estimation evaluates hallucination risk by measuring the consistency among multiple responses generated by the model for a given query (Manakul et al., 2023; Wang et al., 2022). Such response-level consistency signals provide an overall indication of generation reliability, but may not explicitly identify localized factual errors within an otherwise consistent response. To improve fine-grained error localization, recent hallucination detection methods decompose responses into smaller factual units, such as claims, entities, relations, or structured paths, and verify them individually (Hu et al., 2024; Cheng et al., 2025). However, local verification has a complementary weakness: factual units that appear individually supported may still form an inconsistent response when their temporal, relational, or other cross-unit dependencies are considered. As illustrated in Fig. 1, response-level verification may overlook localized factual errors, whereas local verification may miss inconsistencies that emerge only from the broader response context.

More fundamentally, these complementary failure modes reflect a broader limitation of predefined verification strategies. Existing approaches typically predetermine the verification units, contextual scope, and decision procedure, rather than adapting them to the factual requirements of individual samples (Lu et al., 2025). Factual inconsistencies, however, are heterogeneous in both scope and evidence requirements. Some conflicts are localized to a specific entity, number, or concept and therefore require fine-grained evidence checking, whereas others arise from temporal, relational, or semantic dependencies that can only be identified from the broader response context (Zhong & Litman, 2025; Li et al., 2026b). Moreover, different factual elements may require different forms of evidence matching (Tang et al., 2023), and samples can exhibit substantially different levels of decision uncertainty. Applying a predefined verification strategy uniformly across such heterogeneous cases can therefore create a mismatch between the verification procedure and the evidence actually required for reliable factual judgment.

To address these limitations, we introduce Uncertainty-Aware Hierarchical Verification (UAHV), an adaptive hallucination detection framework that adjusts verification according to sample-specific factual states. UAHV incorporates two complementary mechanisms: (i) **Hierarchical Factual Verification (HFV)**, which progressively routes samples from global semantic screening to fine-grained local verification and further sends cases near the local decision boundary to evidence-constrained arbitration. This design allows salient response-level conflicts to be identified without unnecessary fine-grained checking, while reserving stronger verification for samples whose factual state remains uncertain. (ii) **Type-Aware Asymmetric Verification (TAV)**, which applies strict evidence matching to evidence-sensitive facts and context-aware semantic verification to semantically flexible facts. This avoids overly permissive semantic matching for facts that require exact evidence, while reducing false inconsistencies caused by overly rigid lexical matching for semantically flexible content.

Together, HFV and TAV move beyond predefined verification strategies by jointly adapting verification scope, evidence criteria, and reasoning strength to sample-specific factual states and decision uncertainty. This reduces the mismatch between fixed verification procedures and heterogeneous factual conflicts, enabling UAHV to capture both localized and response-level inconsistencies while avoiding unnecessary verification and inappropriate evidence matching. We evaluate UAHV across three benchmarks and four LLM backbones spanning hallucination detection and fact verification. UAHV achieves the highest F1 score in 11 of 12 settings, with an average F1 of 76.12%, exceeding the strongest baseline by 6.27 percentage points.

2 Related Work

2.1 White-box Hallucination Detection

White-box hallucination detection methods exploit model-internal signals and can be broadly categorized into uncertainty-based and representation-based approaches. Uncertainty-based methods estimate hallucination risk from token probabilities, logits, or confidence-related signals (Huang et al., 2023; Fadeeva et al., 2024; Varshney et al., 2023), although such signals do not always faithfully reflect factual correctness (Ji et al., 2025). Representation-based methods instead use hidden activations, attention patterns, or layer-wise inference dynamics to distinguish factual from hallucinated outputs (Azaria & Mitchell, 2023; Jiang et al., 2024; Snyder et al., 2024; Zhu et al., 2024; Liu et al., 2025b). However, these methods require access to model-internal information and do not inherently verify whether generated content is supported by external evidence.

2.2 Black-box Hallucination Detection

Black-box methods mainly include consistency-based and external-knowledge verification approaches. Consistency-based methods detect hallucinations by comparing multiple model outputs or semantically related responses (Zhang et al., 2023; Manakul et al., 2023). External-knowledge methods instead verify generated content against retrieved or trusted evidence. FActScore decomposes generations into atomic facts (Min et al., 2023), HaluCheck verifies decomposed responses against retrieved documents (Heo et al., 2025), and RefChecker represents claims as triplets for fine-grained verification (Hu et al., 2024).

Recent work has further introduced adaptive verification. Bayesian sequential estimation dynamically determines whether additional evidence is required (Wang et al., 2023), while Rowen selectively activates retrieval according to cross-response uncertainty (Ding et al., 2025). However, such adaptivity mainly controls evidence acquisition or retrieval invocation, while verification scope and evidence-matching strategies remain largely predefined.

Unlike prior adaptive methods that mainly adjust evidence acquisition or retrieval invocation, UAHV adapts the verification process itself. It progressively changes verification depth according to the intermediate factual state of each sample, escalating from global screening to fine-grained local verification and boundary arbitration only when necessary. Meanwhile, its type-aware verification applies different evidence-matching criteria to evidence-sensitive and semantically flexible facts. This joint adaptation of verification depth and evidence matching allows UAHV to allocate stronger verification only where needed while avoiding a fixed verification procedure across heterogeneous samples.

3 Methodology

The UAHV framework, as illustrated in Fig. 2, consists of two complementary mechanisms: HFV and TAV. HFV progressively routes samples from coarse-grained global semantic screening to fine-grained local factual verification according to intermediate verification states, and further uses proximity to the local decision boundary as an uncertainty signal to selectively activate evidence-constrained arbitration. Within the local verification stage, TAV applies differentiated verification criteria to heterogeneous factual elements, using strict evidence matching for evidence-sensitive facts while allowing context-aware semantic verification for semantically flexible facts.

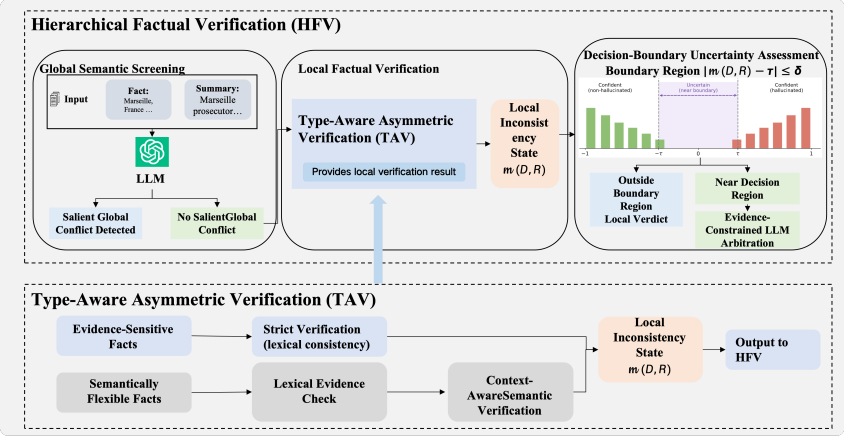


Figure 2: Schematic of the proposed UAHV framework. UAHV integrates HFV and TAV: HFV progressively routes samples from global semantic screening to local verification and boundary-triggered arbitration, while TAV applies differentiated verification strategies to heterogeneous factual elements during local verification.

3.1 PROBLEM SETUP

Given a reference evidence document D and a response to be verified R, reference-grounded hallucination detection aims to determine whether the factual content expressed in R is consistent with the evidence provided by D. We formulate this task as a binary factual-consistency classification problem,

$$\mathcal{H}(D, R) \in \{0, 1\}, \tag{1}$$

where $\mathcal{H}(D, R) = 1$ denotes REFUTES, indicating factual inconsistency with respect to D, and $\mathcal{H}(D, R) = 0$ denotes SUPPORTS. UAHV takes the document–response pair (D, R) as input and predicts the corresponding label $\widehat{\mathcal{H}}(D, R)$. A detailed formulation of the task and factual elements is provided in Appendix A.1.

3.2 HIERARCHICAL FACTUAL VERIFICATION

Existing hallucination detection methods often rely on predefined verification granularity and decision procedures (Zhou et al., 2025), without adapting verification to differences in conflict scope and decision uncertainty across samples. UAHV addresses this limitation through HFV, which progressively routes verification from global semantic screening to type-aware local verification and selectively invokes evidence-constrained arbitration near the local decision boundary. By treating boundary proximity as an operational uncertainty signal, HFV concentrates additional reasoning on uncertain cases rather than applying uniform verification to all samples.

Let D denote the reference evidence and R the text under verification. The global verification gate is defined as

$$g(D, R) = \mathbb{I}[G(D, R) = \text{SUPPORTS}], \tag{2}$$

where $G(D, R) \in \{\text{SUPPORTS}, \text{REFUTES}\}$ is the response-level global verifier and $\mathbb{I}[\cdot]$ is the indicator function. When $g(D, R) = 0$, the global verifier detects an evident response-level factual conflict, and the sample receives a final REFUTES decision. When $g(D, R) = 1$, the sample proceeds to local verification. Thus, a global SUPPORTS prediction serves as a routing signal and remains subject to further checking.

For local verification, each factual element f_i is represented by a distinct keyword extracted from the response R, together with its associated response sentences. These keywords serve as lexical proxies for localized factual content, including evidence-sensitive facts and semantically flexible facts. Keywords are lowercased and deduplicated, so that each retained keyword form contributes at most once to the inconsistency count. Let N denote the number of these elements and $T \in \mathbb{Z}_{\geq 1}$ the local missing-count threshold. The local inconsistency state is defined as

$$m(D, R) = \begin{cases} \sum_{i=1}^{N} \left[1 - \Phi_i(D, R)\right], & g(D, R) = 1, \\ 0, & g(D, R) = 0, \end{cases} \tag{3}$$

where $\Phi_i(D, R) \in \{0, 1\}$ is the local support judgment for f_i, instantiated by TAV in the next subsection. A value of one indicates that the element is accepted by the local verification rules, while zero indicates that it remains unresolved. The dependence on R includes the sample-level condition controlling semantic rescue. Consequently, $m(D, R)$ is an operational estimate of factual inconsistency based on unresolved lexical proxies. For globally rejected samples, it is set to zero by convention because the local decision is unused.

For samples satisfying $g(D, R) = 1$, the local decision is

$$L(D, R) = \begin{cases} \text{SUPPORTS}, & m(D, R) < T, \\ \text{REFUTES}, & m(D, R) \geq T. \end{cases} \tag{4}$$

The local verifier converts the inconsistency count $m(D, R)$ into a binary decision using threshold T. We characterize uncertainty through the stability of this decision rather than through model-internal confidence. When $m(D, R)$ is far from T, small changes in individual element-level support judgments are unlikely to alter the resulting prediction. In contrast, when $m(D, R)$ lies close to the threshold, even a small change in an element-level judgment may move the inconsistency count

across the decision boundary and reverse the prediction. We therefore use proximity to the local decision boundary as an operational proxy for decision uncertainty. This uncertainty signal is used to identify samples requiring additional verification and is not intended as a calibrated probability of error. We further validate this boundary-based uncertainty signal in Appendix A.8, where samples near the local decision boundary are shown to exhibit higher empirical error rates than those farther away.

Because the inconsistency count is integer-valued, the local decision boundary lies between the two adjacent counts T-1 and T. We therefore define the arbitration gate as

$$\beta(D,R) = g(D,R) \cdot \mathbb{I}[\max(0, T-\delta) \le m(D,R) \le T+\delta-1], \tag{5}$$

where $\delta \in \mathbb{Z}_{\ge 1}$ denotes the number of integer count levels included on each side of the local decision boundary.

The routing state of a sample is encoded by

$$\boldsymbol{\psi}(D,R) = \begin{bmatrix} 1-g(D,R) \\ g(D,R)-\beta(D,R) \\ \beta(D,R) \end{bmatrix}. \tag{6}$$

Since $\beta(D,R) \le g(D,R)$ and both variables are binary, the components are binary and sum to one. The states $[1,0,0]^\top$, $[0,1,0]^\top$, and $[0,0,1]^\top$ respectively represent global termination, a direct local decision, and routing to evidence-constrained arbitration.

For a boundary sample, the arbiter receives the complete pair (D,R) together with auxiliary local information, including potentially unsupported keywords, candidate evidence sentences, similarity scores, and the local verification result. These local signals guide the re-examination of suspicious content. The arbiter may confirm or override the local decision. Let $A(D,R) \in \{\mathrm{SUPPORTS}, \mathrm{REFUTES}\}$ denote the effective arbitration outcome, with its auxiliary local input left implicit. If an arbitration result is unavailable, the implementation falls back to $L(D,R)$, which is then used as $A(D,R)$.

$$\widehat{Y}(D,R) = \begin{cases} \mathrm{REFUTES}, & g(D,R) = 0, \\ L(D,R), & g(D,R) = 1 \wedge \beta(D,R) = 0, \\ A(D,R), & \beta(D,R) = 1. \end{cases} \tag{7}$$

HFV therefore realizes the uncertainty-aware hierarchical routing of UAHV by adapting verification scope and reasoning strength to intermediate verification states and decision-boundary uncertainty, reserving additional evidence-constrained reasoning for samples whose local decisions remain uncertain.

3.3 TYPE-AWARE ASYMMETRIC VERIFICATION

Different lexical elements tolerate different degrees of semantic variation. Proper names and numerical expressions often require close lexical correspondence, whereas common nouns may remain supported under paraphrasing (Mujahid et al., 2026; Kryściński et al., 2020). TAV operationalizes this distinction through part-of-speech information and applies semantic rescue selectively to eligible common-noun forms. The resulting support judgments determine the local inconsistency count used by HFV.

Let w_i denote the i-th localized factual element, and let z_i denote its verification profile. Rather than assigning verification strategies solely according to surface-level factual labels, TAV determines z_i according to the semantic role and structural dependency of each element. Specifically, an element is treated as *evidence-sensitive* when its factual validity depends on preserving entity identity, numerical value, relational structure, predicate polarity, or factual bindings, whereas elements that allow semantically equivalent realizations are treated as *semantically flexible*. In the implementation, $z_i = \mathrm{Flexible}$ for common-noun forms $w_i \in \mathcal{N}_R$, while $z_i = \mathrm{Sensitive}$ for the remaining localized elements. Thus, membership in the common-noun set determines semantic-rescue eligibility. All other retained keyword forms use the lexical-only branch.

Let $\mathcal{V}_D$ denote the keyword vocabulary extracted from D using the same procedure. The lexical matching score is

$$s_i^{\text{lex}} = \max_{v \in \mathcal{V}_D} \text{sim}_{\text{lex}}(w_i, v). \tag{8}$$

Here, lexical similarity is measured using a character-level sequence similarity function. An exact lexical match receives a score of one, while the similarity score is defined as zero when $\mathcal{V}_D$ is empty. Given the lexical threshold τ_{lex}, the initial lexical support state is

$$\ell_i = \mathbb{I}\left[s_i^{\text{lex}} \geq \tau_{\text{lex}}\right]. \tag{9}$$

When $\tau_{\text{lex}} < 1$, fuzzy lexical matches are permitted.

Semantic rescue additionally depends on the sample-level lexical inconsistency state. Define the initial missing count as

$$m_{\text{lex}}(D, R) = \sum_{i=1}^{N} (1 - \ell_i), \tag{10}$$

and its rescue activation gate as

$$\eta(D, R) = g(D, R) \cdot \mathbb{I}[m_{\text{lex}}(D, R) \geq T]. \tag{11}$$

The same threshold T is applied to the pre-rescue lexical inconsistency count to determine whether semantic rescue is activated and to the post-rescue inconsistency count to determine the final local decision.

To determine the semantic support state ρ_i for an eligible flexible element, TAV ranks the sentences in D by semantic similarity to each response sentence containing w_i and retrieves the top-K evidence sentences for each such response sentence. Candidate common nouns are then extracted from the retrieved evidence sentences. Let $\mathcal{C}_{K,i}$ denote the resulting candidate set for w_i under the pair (D, R). Each candidate is represented as $c_{ij} = (q_{ij}, d_{ij}, w_{ij})$, where q_{ij} is a response sentence containing w_i, d_{ij} is a retrieved evidence sentence, and w_{ij} is a candidate common noun extracted from d_{ij}. Each retrieved evidence sentence may contribute multiple candidate nouns, and when w_i occurs in multiple response sentences, candidates are collected across all corresponding sentence pairs.

For each candidate, TAV evaluates semantic consistency from three complementary perspectives using the same semantic encoder $E(\cdot)$ and cosine similarity. The sentence-level score compares the complete response sentence q_{ij} with the retrieved evidence sentence d_{ij}:

$$s_{ij}^{\text{sent}} = \cos(E(q_{ij}), E(d_{ij})). \tag{12}$$

The isolated score measures the semantic similarity between the response factual element w_i and the candidate factual element w_{ij} independently of their surrounding contexts:

$$s_{ij}^{\text{iso}} = \cos(E(w_i), E(w_{ij})). \tag{13}$$

To further account for context-dependent meaning, TAV constructs contextualized representations of the factual elements and their corresponding sentences $h(w, s)$ for semantic comparison.

The contextual score is then defined as

$$s_{ij}^{\text{ctx}} = \cos(E(h(w_i, q_{ij})), E(h(w_{ij}, d_{ij}))). \tag{14}$$

Thus, s_{ij}^{sent} measures sentence-level semantic agreement, s_{ij}^{iso} captures the semantic correspondence between the factual elements themselves, and s_{ij}^{ctx} evaluates whether this correspondence is preserved under their respective sentence contexts.

The combined semantic score is

$$s_{ij}^{\text{sem}} = \lambda_{\text{iso}} s_{ij}^{\text{iso}} + \lambda_{\text{ctx}} s_{ij}^{\text{ctx}} + \lambda_{\text{sent}} s_{ij}^{\text{sent}}. \tag{15}$$

A candidate must satisfy all four similarity thresholds. Let $\mathcal{U} = \{\text{sent}, \text{iso}, \text{ctx}, \text{sem}\}$ index the four criteria. Define

$$v_{ij} = \prod_{u \in \mathcal{U}} \mathbb{I}\big[s_{ij}^{u} \geq \tau_u\big], \tag{16}$$

where τ_{sem} is the threshold for the combined score. The semantic support state is

$$\rho_i = \mathbb{I}[\exists\, c_{ij} \in \mathcal{C}_{K,i} : v_{ij} = 1]. \tag{17}$$

Thus, at least one candidate must jointly satisfy the sentence-level, isolated, contextual, and combined thresholds. When $\mathcal{C}_{K,i}$ is empty, $\rho_i = 0$.

The support function used by HFV is

$$\Phi_i(D, R) = \ell_i + (1 - \ell_i)\, \eta(D, R)\, \mathbb{I}[z_i = \text{Flexible}]\, \rho_i. \tag{18}$$

A successful lexical match directly establishes support for the corresponding factual element. For elements without sufficient lexical evidence, the verification strategy depends on their verification profile: evidence-sensitive elements retain the strict lexical decision, whereas semantically flexible elements may receive additional support from semantic evidence. Thus, $\Phi_i(D, R)$ combines strict lexical evidence matching with selective semantic rescue and provides the element-level support judgment used by HFV.

Based on these element-level support states, HFV aggregates unsupported factual elements into the final local inconsistency count:

$$m(D, R) = \begin{cases} \displaystyle\sum_{i=1}^{N} (1 - \ell_i)\, [1 - \eta(D, R) \mathbb{I}[z_i = \text{Flexible}]\, \rho_i], & g(D, R) = 1, \\ 0, & g(D, R) = 0. \end{cases} \tag{19}$$

Accordingly, a lexically unsupported evidence-sensitive element continues to contribute to the inconsistency count, whereas a lexically unsupported semantically flexible element no longer contributes when semantic rescue is activated and sufficient semantic evidence is found. The resulting inconsistency count is then compared with the threshold T to obtain the local decision $L(D, R)$.

TAV applies asymmetric verification to different factual content: evidence-sensitive facts rely on strict evidence matching, while semantically flexible facts may receive semantic verification when lexical evidence is insufficient. These local signals are complemented by global screening and boundary arbitration over the complete document–response pair, enabling broader relational and predicate-level inconsistencies to be captured while preserving appropriate semantic flexibility.

4 Experiments

4.1 Experimental Settings

Datasets. To validate UAHV, we evaluate it on three diverse datasets: HaluEval-Summary (Li et al., 2023) for assessing complex hallucinations in long summaries; QAGS-CNNDM (Honovich et al., 2022) for evaluating factual consistency; and FEVER (Honovich et al., 2022) for atomic-level fact verification. To manage computational overhead, we follow the sampling strategy of G2LDetect (Cheng et al., 2025), using 1,000 source records from HaluEval-Summary, 233 instances from QAGS-CNNDM, and 500 instances from FEVER. Each selected HaluEval-Summary record contains one factually consistent summary and one hallucinated summary, resulting in 2,000 classification instances for the SUM evaluation set. We report macro-averaged Precision, Recall, and F1 scores, together with the false positive rate (FPR). For FPR computation, Hallucination is treated as the positive class.

Baselines. To validate the effectiveness of our method, we compare it with representative hallucination detection baselines from two categories: (1) *End-to-end verification*, where a standard LLM directly assesses factual consistency in a single inference process; and (2) *Fine-grained or structured verification methods*, including SelfCheckGPT (Manakul et al., 2023), FactScore (Min et al., 2023), FacTool (Chern et al., 2023), RefChecker (Hu et al., 2024), and G2LDetect (Cheng et al., 2025). These methods cover sentence-level, atomic-fact, claim-level, tool-augmented, and global-to-local verification paradigms. All baselines follow the recommended settings of their respective publications whenever applicable.

Method	SUM				QAG				FEVER			
	F1↑	P↑	R↑	FPR↓	F1↑	P↑	R↑	FPR↓	F1↑	P↑	R↑	FPR↓
Llama-3-8B-Instruct												
Standard	0.6625	0.6849	*0.6695*	**18.60**	*0.7498*	*0.7767*	*0.7537*	10.26	0.7071	0.6729	0.8506	15.25
SelfCheckGPT	0.6383	0.6536	0.6440	23.10	**0.7854**	0.7857	**0.7855**	23.08	*0.6819*	*0.6499*	*0.8440*	16.59
FactScore	0.4141	0.5441	0.5140	89.90	0.6731	0.6757	0.6740	37.61	0.4811	0.5733	0.7436	46.41
FacTool	0.6800	*0.6832*	0.6810	26.40	0.6897	0.6937	0.6907	24.79	0.6689	0.6444	**0.8654**	19.61
RefChecker	0.5962	0.6769	0.6255	64.40	0.6484	0.6861	0.6601	15.38	0.4880	0.5563	0.6844	41.18
G2LDetect	*0.6642*	0.6696	0.6660	*26.10*	0.6470	0.7061	0.6643	*11.11*	0.6122	0.5949	0.6910	*15.47*
UAHV	**0.7085**	**0.7085**	**0.7085**	29.80	0.7792	**0.8196**	0.7847	**5.13**	**0.7477**	**0.7078**	0.8245	**8.28**
Llama-3.1-8B-Instruct												
Standard	*0.6241*	0.6773	0.6425	57.90	*0.6517*	*0.7442*	0.6447	5.95	0.7098	0.6736	0.8728	15.69
SelfCheckGPT	0.5617	0.5776	0.5710	28.30	0.6049	0.7578	0.6467	**2.56**	*0.6843*	*0.6529*	*0.8641*	*17.43*
FactScore	0.4137	0.6315	0.5285	**2.90**	0.6811	0.6860	0.6827	38.46	0.4557	0.5647	0.7140	49.89
FacTool	0.5513	0.5779	0.5670	62.00	0.5074	0.5774	0.5506	74.36	0.2973	0.5481	0.6153	74.51
RefChecker	0.5749	0.6211	0.5965	62.90	0.6433	0.6825	*0.6558*	15.38	0.4468	0.5481	0.6595	48.58
G2LDetect	0.6355	*0.6355*	*0.6355*	36.00	0.6273	0.6883	0.6471	11.97	0.5908	0.5840	0.7025	20.48
UAHV	**0.7047**	**0.7079**	**0.7055**	*34.80*	**0.7711**	**0.8062**	**0.7762**	*6.84*	**0.7757**	**0.7263**	**0.8833**	**8.71**
Qwen2.5-7B-Instruct												
Standard	0.6807	0.7476	0.6970	**7.70**	0.6762	0.7797	0.6970	3.45	0.7488	*0.7118*	*0.8660*	*14.60*
SelfCheckGPT	0.7091	*0.7295*	0.7140	*15.60*	0.7285	*0.7759*	0.7374	*7.69*	0.5815	**0.7759**	0.5577	**0.65**
FactScore	0.3960	0.5641	0.5140	92.80	0.6793	0.6906	0.6828	41.88	0.4577	0.4728	0.4505	17.21
FacTool	*0.7047*	0.7076	*0.7055*	24.40	*0.7257*	0.7426	*0.7291*	15.38	*0.7020*	0.6646	0.8728	15.69
RefChecker	0.6235	0.6583	0.6365	54.90	0.6885	0.6967	0.6906	22.22	0.5156	0.5745	0.7429	39.22
G2LDetect	0.6807	0.6832	0.6815	27.00	0.6939	0.7326	0.7031	11.97	0.5844	0.5934	0.7674	27.02
UAHV	**0.7398**	**0.7556**	**0.7430**	14.60	**0.7535**	**0.8155**	**0.7631**	**3.42**	**0.7759**	0.7237	**0.9033**	9.59
GPT-4o-mini												
Standard	0.6717	**0.7913**	0.6980	**1.90**	*0.7778*	*0.8283*	*0.7846*	**3.42**	0.7224	0.6797	*0.9015*	14.81
SelfCheckGPT	0.7120	0.7476	0.7200	11.30	0.8350	**0.8520**	0.8365	5.98	0.7821	*0.7281*	0.9155	*9.59*
FactScore	0.3927	0.4052	0.4260	80.80	0.5035	0.5403	0.5312	23.08	0.7120	**0.7592**	0.6821	**2.61**
FacTool	*0.7119*	0.7290	*0.7160*	*16.50*	0.7518	0.7699	0.7549	12.82	*0.7491*	0.7008	0.8835	11.11
RefChecker	0.6998	0.7042	0.7010	36.20	0.7667	0.7748	0.7679	15.38	0.6030	0.6090	0.8161	27.02
G2LDetect	0.6011	0.6245	0.6115	55.00	0.6860	0.7198	0.6946	13.68	0.6373	0.6183	0.7833	18.95
UAHV	**0.7308**	*0.7317*	**0.7310**	24.20	**0.8365**	0.8394	**0.8367**	*11.97*	**0.8112**	0.7543	**0.9353**	8.06

Table 1: Performance comparison across different backbone models. F1, precision (P), and recall (R) are reported as decimal values, and FPR as a percentage (%). Best, second-best, and third-best results within each backbone are shown in **bold**, underline, and *italics*, respectively.

4.2 MAIN RESULTS

Comparison with Hallucination Detection Methods. As shown in Table 1, UAHV consistently outperforms existing hallucination detection methods across different backbone models and datasets, achieving the best F1 in 11 of the 12 settings and ranking second in the remaining one. On average, UAHV improves F1 from 0.6986 for the strongest baseline to 0.7612, corresponding to an absolute gain of 0.0627. The consistent improvements across LLaMA, Qwen, and GPT-4o-mini further indicate that the effectiveness of UAHV is not tied to a particular backbone model. Notably, UAHV also achieves strong F1 and recall with relatively controlled FPR on QAG and FEVER. Although a higher FPR is observed on SUM, the corresponding gains in overall F1 indicate a different precision–recall trade-off rather than a general degradation in verification quality.

The results also highlight the limitations of using a uniform verification strategy. Standard prompting relies primarily on response-level reasoning and may miss localized factual inconsistencies, while methods based on repeated or fixed verification procedures do not explicitly adapt verification strength to different factual risks or uncertainty levels. In contrast, UAHV progressively combines global semantic screening, type-aware local verification, and uncertainty-triggered arbitration, allowing both verification granularity and reasoning strength to vary according to the intermediate verification state. The asymmetric treatment of evidence-sensitive and semantically flexible facts further avoids overly rigid lexical matching while preserving strict verification for factual elements whose identity or numerical value must be maintained.

4.3 ABLATION STUDIES

We conduct ablation studies to investigate the contribution of semantic rescue and boundary arbitration in UAHV using Llama-3.1-8B-Instruct. As shown in Table 2, semantic rescue consistently improves or maintains detection performance across the three datasets, with more evident gains on SUM and FEVER. This suggests that context-aware semantic verification can recover flexible factual expressions that would otherwise be incorrectly rejected by strict lexical matching. The improvement is relatively limited on QAG, indicating that its factual elements rely less on semantic recovery under the current verification setting. Boundary arbitration provides a further complementary effect by reconsidering samples close to the local decision boundary, where the preceding verification stages are more likely to remain uncertain. Although its contribution varies with the distribution of ambiguous samples across datasets, combining both components yields the strongest overall performance, with Full UAHV achieving F1 scores of 0.7047, 0.7711, and 0.7757 on SUM, QAG, and FEVER, respectively. On FEVER, the complete framework also reduces FPR from 10.02% to 8.71%. These results indicate that semantic rescue and boundary arbitration address different sources of verification error: the former mitigates overly rigid evidence matching, whereas the latter refines uncertain local decisions, making their combination complementary within the progressive verification framework.

Dataset	Metric	Global+Lexical	+Rescue	+Arbiter	Full UAHV
SUM	F1 ↑	0.6857	0.6994	*0.6979*	**0.7047**
	P ↑	**0.7088**	*0.6998*	0.6981	0.7079
	R ↑	0.6920	0.6995	*0.6980*	**0.7055**
	FPR ↓	**16.60**	28.00	*28.90*	34.80
QAG	F1 ↑	0.7663	0.7663	*0.7616*	**0.7711**
	P ↑	0.8033	0.8033	*0.8003*	**0.8062**
	R ↑	*0.7718*	0.7719	0.7675	**0.7762**
	FPR ↓	**6.84**	**6.84**	**6.84**	**6.84**
FEVER	F1 ↑	0.7573	*0.7633*	0.7663	**0.7757**
	P ↑	0.7089	*0.7144*	0.7173	**0.7263**
	R ↑	0.8767	*0.8789*	0.8800	**0.8833**
	FPR ↓	10.02	*9.59*	9.37	**8.71**

Table 2: Ablation study of UAHV using Llama-3.1-8B-Instruct. F1, P, and R are reported as decimals and FPR as a percentage (%). REFUTES/Hallucination is treated as the positive class. Best, second-best, and third-best results are shown in **bold**, underline, and *italics*, respectively.

4.4 FURTHER ANALYSIS

Due to space constraints, analyses on computational efficiency, parameter sensitivity, semantic rescue, boundary arbitration, boundary uncertainty, type-aware verification, and case studies are provided in Appendix A.4–A.10.

5 CONCLUSION

This study introduces UAHV, an uncertainty-aware hierarchical verification framework for hallucination detection in large language models. UAHV combines HFV, which progressively routes verification from global semantic screening to fine-grained local verification and uses proximity to the local decision boundary as an uncertainty signal to dynamically trigger evidence-constrained arbitration, with TAV, which applies asymmetric verification to evidence-sensitive and semantically flexible factual elements. This design enables verification depth and strategy to adapt to factual risk and decision uncertainty. Experiments on SUM, QAG, and FEVER across multiple LLM backbones show that UAHV outperforms existing baselines in most settings, demonstrating the effectiveness and robustness of hierarchical verification, type-aware verification, and uncertainty-driven arbitration.

5.1 AI USE STATEMENT

In this work, we used generative AI tools to assist with language polishing, LaTeX formatting, and code development and debugging. Generative AI was also used to help improve the clarity and presentation of the manuscript. All AI-assisted text and code were reviewed and revised by the authors.

Experimental results reported in this paper were obtained by running our own implementations and were manually checked for correctness. The authors take full responsibility for the final content of the paper, including all claims, experimental results, and artifacts produced with the assistance of generative AI.

Reproducibility Statement

We provide detailed implementation and experimental information to support the reproducibility of our results. The model backbones, datasets, evaluation metrics, verification procedures, and implementation details of UAHV are described in the main paper and appendix. We additionally report the fixed hyperparameter configurations used for each benchmark and provide further analyses of the main components and decision mechanisms. An anonymized code package is included in the supplementary materials, containing the implementation of UAHV, experiment configurations, execution interfaces, and logged experimental results. Dataset preprocessing and evaluation follow the settings described in the paper and supplementary materials.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Gabriel Aracena, Kyle Luster, Fabio Santos, Igor Steinmacher, and Marco Aurelio Gerosa. Applying large language models to issue classification. In *Proceedings of the Third ACM/IEEE International Workshop on NL-based Software Engineering*, pp. 57–60, 2024.

Amos Azaria and Tom Mitchell. The internal state of an llm knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 967–976, 2023.

Yejin Bang, Ziwei Ji, Alan Schelten, Anthony Hartshorn, Tara Fowler, Cheng Zhang, Nicola Cancedda, and Pascale Fung. Hallulens: Llm hallucination benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 24128–24156, 2025.

Xiaoxia Cheng, Zeqi Tan, Zhe Zheng, and Weiming Lu. G2ldetect: A global-to-local approach for hallucination detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 102–109, 2025.

I Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, Pengfei Liu, et al. Factool: Factuality detection in generative ai–a tool augmented framework for multi-task and multi-domain scenarios. *arXiv preprint arXiv:2307.13528*, 2023.

Hanxing Ding, Liang Pang, Zihao Wei, Huawei Shen, and Xueqi Cheng. Rowen: Adaptive retrieval-augmented generation for hallucination mitigation in llms. In *Proceedings of the 2025 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, pp. 12–21, 2025.

Ekaterina Fadeeva, Aleksandr Rubashevskii, Artem Shelmanov, Sergey Petrakov, Haonan Li, Hamdy Mubarak, Evgenii Tsymbalov, Gleb Kuzmin, Alexander Panchenko, Timothy Baldwin, et al. Fact-checking the output of large language models via token-level uncertainty quantification. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 9367–9385, 2024.

Yuxin Fan, Yuxiang Wang, Lipeng Liu, Xirui Tang, Na Sun, and Zidong Yu. Research on the online update method for retrieval-augmented generation (rag) model with incremental learning. In *2025 5th International Conference on Neural Networks, Information and Communication Engineering (NNICE)*, pp. 1740–1744. IEEE, 2025.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Anisha Gunjal and Greg Durrett. Molecular facts: Desiderata for decontextualization in llm fact verification. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 3751–3768, 2024.

Sangwoo Heo, Sungwook Son, and Hyunwoo Park. Halucheck: Explainable and verifiable automation for detecting hallucinations in llm responses. *Expert Systems with Applications*, 272:126712, 2025.

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. True: Re-evaluating factual consistency evaluation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 3905–3920, 2022.

Xiangkun Hu, Dongyu Ru, Lin Qiu, Qipeng Guo, Tianhang Zhang, Yang Xu, Yun Luo, Pengfei Liu, Yue Zhang, and Zheng Zhang. Refchecker: Reference-based fine-grained hallucination checker and benchmark for large language models. *arXiv preprint arXiv:2405.14486*, 2024.

Yuheng Huang, Jiayang Song, Zhijie Wang, Shengming Zhao, Huaming Chen, Felix Juefei-Xu, and Lei Ma. Look before you leap: An exploratory study of uncertainty measurement for large language models. *arXiv preprint arXiv:2307.10236*, 2023.

Binyuan Hui, Jian Yang, Zeyu Cui, Jiaxi Yang, Dayiheng Liu, Lei Zhang, Tianyu Liu, Jiajun Zhang, Bowen Yu, Keming Lu, et al. Qwen2. 5-coder technical report. *arXiv preprint arXiv:2409.12186*, 2024.

Ziwei Ji, Lei Yu, Yeskendir Koishekenov, Yejin Bang, Anthony Hartshorn, Alan Schelten, Cheng Zhang, Pascale Fung, and Nicola Cancedda. Calibrating verbal uncertainty as a linear feature to reduce hallucinations. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 3769–3793, 2025.

Che Jiang, Biqing Qi, Xiangyu Hong, Dayuan Fu, Yang Cheng, Fandong Meng, Mo Yu, Bowen Zhou, and Jie Zhou. On large language models' hallucination with regard to known facts. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 1041–1053, 2024.

Linggang Kong, Yunlong Zhang, Xiaofeng Zhong, Haoran Fu, Yongjie Wang, and Huijun Liu. Halugnn: Hallucination detection in large language models using graph neural network. *Expert Systems with Applications*, pp. 130857, 2025.

Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. Evaluating the factual consistency of abstractive text summarization. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 9332–9346, 2020.

Jean Lee, Nicholas Stevens, and Soyeon Caren Han. Large language models in finance (finllms). *Neural Computing and Applications*, 37(30):24853–24867, 2025.

Junyi Li, Xiaoxue Cheng, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. Halueval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 6449–6464, 2023.

Wenyun Li, Zheng Zhang, Dongmei Jiang, and Xiangyuan Lan. Bolster hallucination detection via prompt-guided data augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 37618–37626, 2026a.

Zituo Li, Guangzhou Chen, Jianbin Sun, Qi Song, and Yang Kewei. Hallucination detection in long-form text generated by llms: A benchmark and a hyper-relational knowledge graph approach. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 33477–33494, 2026b.

Mushui Liu, Yuhang Ma, Zhen Yang, Jun Dan, Yunlong Yu, Zeng Zhao, Zhipeng Hu, Bai Liu, and Changjie Fan. Llm4gen: Leveraging semantic representation of llms for text-to-image generation. In *Proceedings of the AAAI conference on Artificial Intelligence*, volume 39, pp. 5523–5531, 2025a.

Qiang Liu, Xinlong Chen, Yue Ding, Bowen Song, Weiqiang Wang, Shu Wu, and Liang Wang. Attention-guided self-reflection for zero-shot hallucination detection in large language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 21016–21032, 2025b.

Yining Lu, Noah Ziems, Hy Dang, and Meng Jiang. Optimizing decomposition for optimal claim verification. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 5095–5114, 2025.

Potsawee Manakul, Adian Liusie, and Mark Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 9004–9017, 2023.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 12076–12100, 2023.

Zain Muhammad Mujahid, Dustin Wright, and Isabelle Augenstein. Stress testing factual consistency metrics for long-document summarization. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 31914–31933, 2026.

Ben Snyder, Marius Moisescu, and Muhammad Bilal Zafar. On early detection of hallucinations in factual question answering. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2721–2732, 2024.

Liyan Tang, Tanya Goyal, Alex Fabbri, Philippe Laban, Jiacheng Xu, Semih Yavuz, Wojciech Kryściński, Justin Rousseau, and Greg Durrett. Understanding factual errors in summarization: Errors, summarizers, datasets, error detectors. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11626–11644, 2023.

Neeraj Varshney, Wenlin Yao, Hongming Zhang, Jianshu Chen, and Dong Yu. A stitch in time saves nine: Detecting and mitigating hallucinations of llms by validating low-confidence generation. *arXiv preprint arXiv:2307.03987*, 2023.

Xiaohua Wang, Yuliang Yan, Longtao Huang, Xiaoqing Zheng, and Xuan-Jing Huang. Hallucination detection for generative large language models by bayesian sequential estimation. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pp. 15361–15371, 2023.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*, 2022.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*, 2023.

Jiaxin Zhang, Zhuohang Li, Kamalika Das, Bradley Malin, and Sricharan Kumar. Sac3: reliable hallucination detection in black-box language models via semantic-aware cross-check consistency. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 15445–15458, 2023.

Yang Zhong and Diane Litman. Discourse-driven evaluation: Unveiling factual inconsistency in long document summarization. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 2050–2073, 2025.

Xiaoling Zhou, Mingjie Zhang, Zhemg Lee, Wei Ye, and Shikun Zhang. Hademif: Hallucination detection and mitigation in large language models. In *International Conference on Learning Representations*, volume 2025, pp. 81014–81038, 2025.

Derui Zhu, Dingfan Chen, Qing Li, Zongxiong Chen, Lei Ma, Jens Grossklags, and Mario Fritz. Pollmgraph: Unraveling hallucinations in large language models via state transition dynamics. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pp. 4737–4751, 2024.

A APPENDIX

A.1 PROBLEM SETUP

Reference-Grounded Hallucination Detection. Given a reference evidence document D and a response to be verified R, the goal of hallucination detection is to determine whether the factual content expressed in R is consistent with the evidence provided by D. Here, R denotes the text under verification and can correspond either to a model-generated response or to a factual claim, depending on the original task formulation.

We denote the factual elements expressed in R as

$$\mathcal{F} = \{f_1, f_2, \ldots, f_N\}, \tag{20}$$

where N is the number of localized factual elements and each f_j may correspond to a named entity, numerical expression, semantically flexible concept, or factual relation. A factual element is considered locally inconsistent when it is insufficiently supported by or contradictory to the reference evidence D.

We formulate hallucination detection as a binary factual-consistency classification problem, with the target label defined as

$$\mathcal{H}(D, R) \in \{0, 1\}, \tag{21}$$

where $\mathcal{H}(D, R) = 1$ denotes REFUTES, indicating factual inconsistency with respect to D, and $\mathcal{H}(D, R) = 0$ denotes SUPPORTS. UAHV estimates this target label through global semantic screening, type-aware local verification, and uncertainty-triggered arbitration. Its predicted label is denoted by $\widehat{\mathcal{H}}(D, R)$.

Hallucination detection and fact verification share the common objective of assessing factual consistency between a text and supporting evidence. We therefore include FEVER as an auxiliary factual-consistency benchmark, where R denotes the claim and D its associated evidence, to evaluate whether UAHV generalizes beyond generation-oriented settings.

UAHV is not limited to entity-level inconsistencies. Local verification handles different factual elements with type-aware evidence matching, while global screening and evidence-constrained arbitration operate on the complete (D, R) pair to capture broader relational and semantic conflicts.

Overall, UAHV progressively verifies factual consistency from global semantics to localized evidence and selectively strengthens verification for uncertain samples.

A.2 IMPLEMENTATION AND REPRODUCIBILITY DETAILS

Backbone Models and Inference. We conduct experiments with LLaMA-3-8B-Instruct, LLaMA-3.1-8B-Instruct (Grattafiori et al., 2024), Qwen2.5-7B-Instruct(Hui et al., 2024), and GPT-4o-mini(Achiam et al., 2023). The three open-weight models are loaded from their official Hugging Face checkpoints and evaluated in bfloat16 precision, while GPT-4o-mini is accessed through the OpenAI API. Unless otherwise specified, all generative verification stages use deterministic decoding with temperature 0.0. Experiments with open-weight models are conducted on a single NVIDIA A100-PCIE-40GB GPU.

Semantic Verification. Semantic verification is implemented using `all-mpnet-base-v2` sentence embeddings. For each lexically unsupported semantically flexible factual element, UAHV retrieves the top-K candidate evidence sentences and evaluates sentence-level, isolated, contextual, and combined semantic similarities. Hyperparameters are selected from a predefined search space shared across backbone models, as summarized in Table 3.

Boundary Arbitration. A sample is routed to the evidence-constrained arbiter when its local inconsistency count lies near the decision boundary, $T - \delta \leq m(D, R) \leq T + \delta - 1$. Unless otherwise specified, we use $\delta = 1$, corresponding to the two nearest integer states $T - 1$ and T on opposite sides of the boundary. The arbiter receives the reference evidence and the text under verification and returns a binary factual-consistency decision under deterministic decoding.

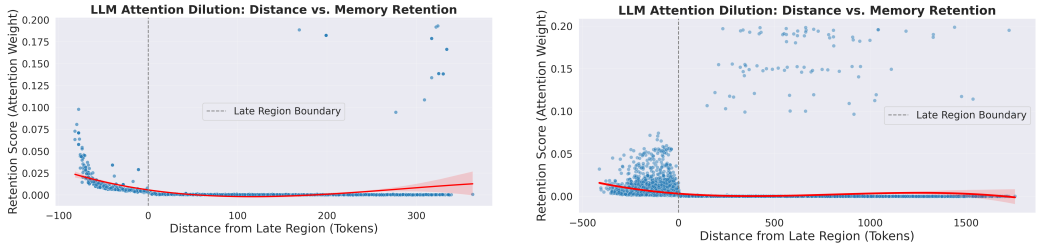
Reproducibility. All dataset subsets are fixed before evaluation and shared across compared methods. HaluEval-Summary uses 1,000 source records, corresponding to 2,000 classification instances,

while QAG and FEVER contain 233 and 500 instances, respectively. Parameter-sensitivity and supplementary analyses use the same subsets as their corresponding main experiments. The exact configurations used for individual runs are retained in the accompanying implementation, and the complete global-verification and arbitration prompt templates are provided below.

Hyperparameter	**Search Range**	**Default**
Fuzzy threshold	$\{0.75, 0.80, 0.85\}$	0.75
Missing-count threshold	$\{2, 3, 4\}$	4
Sentence similarity	$\{0.45, 0.50, 0.55, 0.60\}$	0.45
Isolated similarity	$\{0.50, 0.55, 0.60, 0.65\}$	0.50
Contextual similarity	$\{0.60, 0.65, 0.70, 0.75\}$	0.60
Combined similarity	$\{0.60, 0.63, 0.67, 0.70\}$	0.60
Candidate sentences K	$\{3, 5, 8\}$	3

Table 3: Predefined hyperparameter search space and default configuration used in UAHV. K denotes the maximum number of candidate evidence sentences retrieved for semantic verification.

A.3 Preliminary Analysis of Contextual Evidence Accessibility



(a) Relation retention under increasing contextual distance on QAGS-CNNDM using LLaMA-3-8B.

(b) Relation retention under increasing contextual distance on HaluEval-Summary using LLaMA-3-8B.

Figure 3: Relation retention patterns under increasing contextual distance on QAGS-CNNDM and HaluEval-Summary using LLaMA-3-8B.

To provide additional motivation for heterogeneous verification requirements, we conduct a preliminary analysis on QAGS-CNNDM and HaluEval-Summary. As shown in Fig. 3, relational information exhibits different retention patterns depending on its contextual position. In particular, relational information located farther from the late context tends to receive weaker attention from subsequent tokens, suggesting that factual evidence may exhibit heterogeneous accessibility across contextual positions. Such differences may contribute to different verification difficulties across factual components. These observations provide complementary motivation for adapting verification strategies to heterogeneous factual evidence rather than applying an identical verification criterion to all factual elements.

A.4 Time Complexity and Empirical Efficiency Analysis

Let M denote the number of factual elements extracted from a response, L the number of factual elements identified in the reference document, and S the number of document sentences. Let $Q \leq M$ denote the number of semantically flexible factual elements whose lexical evidence is insufficient and therefore require additional context-aware semantic verification. The computational cost of UAHV mainly arises from four components: global semantic verification, local factual verification, semantic evidence retrieval, and uncertainty-triggered arbitration.

Theoretical complexity. UAHV first performs one response-level global semantic verification for each input. Let C_G denote the cost of this LLM-based verification. Only samples satisfying $g(D, R) = 1$ are subsequently routed to local verification. During this stage, the factual elements extracted from the response are compared against factual evidence from the reference, resulting in a basic matching complexity of

$$O(ML), \tag{22}$$

For evidence-sensitive factual elements, TAV additionally checks applicable relational, polarity, and binding constraints. Since these checks are performed jointly with the corresponding factual matches and introduce only constant-time structural operations for each candidate pair, their cost is absorbed into the $O(ML)$ term.

For the Q semantically flexible elements that remain unsupported by lexical evidence, TAV performs semantic evidence retrieval over the S sentences of the reference document. Given an embedding dimension d, sentence-level similarity computation requires approximately

$$O(QSd), \tag{23}$$

When all candidate sentences are explicitly ranked, candidate ranking further incurs

$$O(QS \log S), \tag{24}$$

Semantic verification is then restricted to the top-K retrieved candidates. Let C denote the average number of candidate concepts examined within each retrieved sentence. The additional isolated and contextual semantic comparisons therefore require approximately

$$O(QKCd), \tag{25}$$

Accordingly, the local verification complexity of a sample that reaches this stage can be expressed as

$$C_{\text{local}} = O(ML + QSd + QS \log S + QKCd)\,, \tag{26}$$

Importantly, this cost is not uniformly incurred by all samples. Under HFV, local verification is activated only when $g(D, R) = 1$, while semantic verification is further restricted to semantically flexible elements whose lexical evidence is insufficient. In practice, this typically yields $Q \ll M$.

Finally, evidence-constrained arbitration is invoked only when the local inconsistency state lies within the uncertainty region around the decision boundary. Let C_A denote the cost of this additional LLM-based verification. Using the boundary activation function $\beta(D, R)$, the per-sample inference cost can therefore be written as

$$O(C_G + g(D, R)\,[ML + QSd + QS \log S + QKCd] + \beta(D, R)C_A)\,, \tag{27}$$

At the dataset level, the expected computational cost can be written more explicitly as

$$\mathbb{E}[C] \approx C_G + P(g = 1)\,\mathbb{E}\,[C_{\text{local}} \mid g = 1] + P(\beta = 1)C_A, \tag{28}$$

where $P(g = 1)$ corresponds to the empirical local-routing rate and $P(\beta = 1)$ corresponds to the boundary-arbitration trigger rate. This formulation highlights that UAHV does not uniformly apply all verification stages to every input, but conditionally allocates additional computation according to the intermediate factual state.

Empirical efficiency. To examine whether this conditional computation structure also translates into practical efficiency, we measure wall-clock inference latency on 100 instances from each benchmark using LLaMA-3.1-8B-Instruct. All methods are evaluated with batch size 1 and three repeated runs. Table 4 reports the average latency per sample together with the average number of generative LLM calls, or prompt-level checks for SelfCheckGPT.

The empirical results are consistent with the conditional computation structure derived above. UAHV requires only 1.35, 1.08, and 1.03 generative LLM calls per sample on SUM, QAG, and FEVER, respectively. These values closely correspond to the observed boundary-arbitration trigger

Dataset	Method	Latency (s/sample) ↓	Calls / Checks per Sample ↓	Speedup vs. G2LDetect
SUM	UAHV	2.826 ± 0.017	1.35	$4.64\times$
	G2LDetect	13.121 ± 0.068	5.07	$1.00\times$
	SelfCheckGPT	0.609 ± 0.094	2.77	$21.53\times$
QAG	UAHV	1.569 ± 0.013	1.08	$4.27\times$
	G2LDetect	6.702 ± 0.026	3.11	$1.00\times$
	SelfCheckGPT	0.501 ± 0.045	3.03	$13.37\times$
FEVER	UAHV	1.228 ± 0.008	1.03	$10.32\times$
	G2LDetect	12.665 ± 0.075	6.94	$1.00\times$
	SelfCheckGPT	0.387 ± 0.007	3.00	$32.75\times$

Table 4: Inference efficiency of UAHV, G2LDetect, and SelfCheckGPT on SUM, QAG, and FEVER using LLaMA-3.1-8B-Instruct. Latency is reported as mean $\pm$ standard deviation over three runs. Calls denotes the average number of generative LLM calls per sample for UAHV and G2LDetect; for SelfCheckGPT, it denotes prompt-level consistency checks and is therefore not strictly equivalent in computational cost.

rates of 35%, 8%, and 3%. Since every sample undergoes one global LLM verification while an additional LLM call is required only when boundary arbitration is activated, the expected number of generative LLM calls satisfies

$$\mathbb{E}[N_{\mathrm{LLM}}] = 1 + \Pr(\beta = 1). \tag{29}$$

which yields 1.35, 1.08, and 1.03 for the three benchmarks, respectively. This close correspondence provides direct empirical evidence that HFV selectively allocates additional LLM reasoning rather than applying a second-stage LLM verifier uniformly to all inputs.

The amount of non-generative local verification nevertheless varies across datasets. The local-routing rates are 74% on SUM, 62% on QAG, and 95% on FEVER. Therefore, a low number of LLM calls should not be interpreted as implying that no additional computation is performed. Samples routed to the local stage still incur lexical matching, structural consistency checking, semantic retrieval, and embedding-based similarity computation, corresponding to the C_{local} term above. This distinction also explains why the number of LLM calls alone does not determine end-to-end wall-clock latency.

Compared with G2LDetect, UAHV substantially reduces both generative LLM calls and inference latency. Specifically, UAHV is approximately $4.64\times$, $4.27\times$, and $10.32\times$ faster on SUM, QAG, and FEVER, respectively. This reduction is especially pronounced on FEVER, where G2LDetect requires an average of 6.94 LLM calls per sample, compared with only 1.03 for UAHV. These results indicate that hierarchical routing can avoid a large fraction of repeated generative reasoning required by more uniformly structured verification procedures.

SelfCheckGPT exhibits the lowest absolute wall-clock latency across the three benchmarks despite performing more prompt-level checks per sample. This observation does not contradict the complexity formulation above, because individual LLM invocations need not have identical computational cost. End-to-end latency depends on input length, generated output length, and the additional non-generative operations performed by each method. SelfCheckGPT primarily performs lightweight prompt-level consistency checks, whereas UAHV additionally conducts factual extraction, local lexical verification, structural consistency checking, and selective semantic evidence retrieval. Therefore, UAHV should be interpreted as providing a different accuracy–efficiency trade-off rather than as minimizing absolute inference latency.

Overall, the theoretical and empirical analyses consistently show that UAHV reduces expensive repeated generative reasoning through selective routing. Fine-grained local verification is conditionally activated, semantic matching is restricted to a subset of flexible factual elements, and additional LLM arbitration is invoked only for samples near the local decision boundary. Consequently, UAHV achieves substantially lower inference cost than G2LDetect while retaining the stronger verification capacity provided by its hierarchical and type-aware verification process.

Parameter	SUM	QAG	FEVER
Fuzzy matching threshold	0.0049	0.0000	0.0032
Missing-count threshold	**0.0155**	**0.0095**	**0.0506**
Sentence similarity threshold	0.0000	0.0000	0.0000
Isolated similarity threshold	0.0020	0.0000	0.0032
Contextual similarity threshold	0.0034	0.0000	0.0063
Combined similarity threshold	0.0010	0.0000	0.0063
Number of candidate sentences K	0.0000	0.0000	0.0000

Table 5: Parameter sensitivity of UAHV across SUM, QAG, and FEVER. Each value reports the maximum variation in Macro-F1 (Δ Macro-F1) over the evaluated parameter range.

Dataset	F1 w/o	F1 w/	ΔF1	FPR w/o	FPR w/	Rescue Trigger	Pred. Changes	Corrected/Harmed
SUM	0.6776	0.6785	+0.0009	45.70	44.70	42.60% (852/2000)	20	10 / 10
QAG	0.7582	0.7582	0.0000	8.55	8.55	1.29% (3/233)	0	0 / 0
FEVER	0.7573	**0.7633**	+0.0060	10.02	9.59	3.00% (15/500)	2	2 / 0

Table 6: Semantic rescue analysis across SUM, QAG, and FEVER. Boundary arbitration is disabled in both variants.

A.5 Parameter Sensitivity Studies

To examine the robustness of UAHV across different factual-consistency benchmarks, we conduct one-factor-at-a-time (OFAT) sensitivity analysis on SUM, QAG, and FEVER. For each experiment, one parameter is varied while all remaining parameters are fixed to the corresponding configuration used in the main experiment. Table 5 reports the maximum Macro-F1 variation observed over the evaluated range.

As shown in Table 5, UAHV is generally stable with respect to the semantic verification parameters across all three benchmarks. Variations in sentence-level, isolated, contextual, and combined semantic similarity thresholds lead to only limited changes in Macro-F1, and changing the number of candidate evidence sentences does not alter the final performance within the evaluated range.

In contrast, the missing-count threshold consistently exhibits the largest influence, with Macro-F1 variations of 0.0155, 0.0095, and 0.0506 on SUM, QAG, and FEVER, respectively. This behavior is expected because the missing-count threshold directly controls the aggregation of unsupported local factual elements and consequently determines whether a sample approaches the local decision boundary. The results therefore suggest that UAHV is relatively insensitive to moderate changes in semantic matching thresholds, whereas the local inconsistency threshold plays a more central role in controlling verification sensitivity and downstream routing.

A.6 Semantic Rescue Analysis

We further isolate the contribution of semantic rescue within type-aware local verification. Boundary arbitration is disabled in this comparison so that the effect of context-aware semantic evidence can be separated from the final arbitration stage.

As shown in Table 6, semantic rescue has a substantially smaller and more dataset-dependent effect than boundary arbitration. On FEVER, semantic rescue improves Macro-F1 by 0.0060 and corrects two erroneous predictions without harming any correct prediction. In contrast, QAG exhibits no sample-level prediction changes, while SUM shows only a marginal Macro-F1 improvement of 0.0009 despite a considerably larger number of samples entering the semantic verification procedure.

A closer examination of SUM further illustrates the conservative nature of this mechanism. Although 852 samples satisfy the conditions for semantic verification, valid rescued terms are identified in only 76 samples and only 20 sample-level predictions change. Among these changes, ten correct previous errors while ten harm previously correct predictions, resulting in no net change in accuracy. Semantic rescue should therefore not be interpreted as the primary source of UAHV's overall performance improvement.

Dataset	F1 w/o Arb.	F1 w/ Arb.	ΔF1	FPR w/o Arb.	FPR w/ Arb.	Trigger Rate	Corrected/Harmed	Net Corr.
SUM	0.6785	**0.7047**	+0.0262	44.70	**34.80**	34.05% (681/2000)	162 / 119	+43
QAG	0.7582	**0.7711**	+0.0129	8.55	**6.84**	5.15% (12/233)	3 / 0	+3
FEVER	0.7633	**0.7757**	+0.0124	9.59	**8.71**	3.00% (15/500)	5 / 1	+4

Table 7: Boundary arbitration analysis across SUM, QAG, and FEVER. FPR is reported in percentage points, with REFUTES treated as the positive class.

Instead, its role is to provide semantic flexibility within TAV. Strict factual elements remain governed by evidence-sensitive verification, whereas semantically flexible elements can receive additional contextual evidence when lexical matching is insufficient. The varying contribution across SUM, QAG, and FEVER indicates that the utility of semantic rescue depends on the prevalence and nature of lexical-semantic variation in each benchmark.

A.7 Boundary Arbitration Analysis

We next investigate the contribution of uncertainty-triggered boundary arbitration across the three benchmarks. For this comparison, the global and local verification stages are kept unchanged, and only the final boundary arbitration mechanism is disabled or enabled.

Table 7 shows a consistent positive contribution from boundary arbitration across all three benchmarks. Macro-F1 improves by 0.0262 on SUM, 0.0129 on QAG, and 0.0124 on FEVER. At the same time, FPR decreases by 9.90, 1.71, and 0.87 percentage points, respectively. Thus, the benefit of arbitration is not limited to a single dataset or factual-consistency setting.

The proportion of samples requiring arbitration, however, differs substantially across datasets. Only 3.00% of FEVER samples and 5.15% of QAG samples reach the arbitration stage, whereas 34.05% of SUM samples fall within the local uncertainty region. This difference indicates that the proposed mechanism should be interpreted as *dynamically triggered* rather than necessarily operating on a uniformly small subset. In particular, summarization exhibits considerably more ambiguous local factual evidence and consequently requires stronger downstream verification more frequently.

The correction behavior also differs across benchmarks. On QAG, all three overridden predictions correct previous errors, whereas on FEVER five of six overrides are beneficial. SUM contains substantially more ambiguous cases: 162 decisions are corrected while 119 previously correct decisions are harmed, resulting in 43 net corrections. Despite this higher ambiguity, arbitration still produces a positive net effect and substantially reduces false-positive hallucination predictions. Overall, these results show that boundary arbitration consistently improves the preceding local decisions, while the amount of intervention required depends on the uncertainty structure of each dataset.

A.8 Boundary Uncertainty Validation

To examine whether the local decision boundary effectively captures prediction uncertainty, we analyze the error distribution of samples at different distances from the boundary. Specifically, boundary arbitration is disabled to avoid its influence on the underlying prediction errors, and routed samples are grouped according to their distance from the missing-count decision threshold. Samples immediately adjacent to the decision boundary are treated as the boundary group, while the remaining routed samples are regarded as farther-away samples. We then compare their empirical error risks across SUM, QAG, and FEVER.

As shown in Table 8, samples located near the local decision boundary consistently exhibit higher error rates than those farther away. On SUM, the error rate increases from 30.60% for farther-away samples to 40.53% for boundary samples, corresponding to a risk ratio of 1.32 and an absolute risk increase of 9.93 percentage points. A similar tendency is observed on QAG, where boundary samples exhibit an error rate of 50.00%, compared with 28.67% for farther-away samples. Although the boundary subset on QAG contains only 12 samples and should therefore be interpreted cautiously, its error risk is still 1.74 times that of the farther-away group.

The distinction is particularly pronounced on FEVER. Boundary samples yield an error rate of 40.00%, whereas the error rate among farther-away samples is only 2.86%. This corresponds to a

Dataset	N_{bd}	Boundary Err.	Farther Err.	RR	Δ Risk
SUM	681	40.53%	30.60%	1.32×	+9.93 pp
QAG	12	50.00%	28.67%	1.74×	+21.33 pp
FEVER	15	40.00%	2.86%	13.97×	+37.14 pp

Table 8: Boundary uncertainty validation across SUM, QAG, and FEVER. Boundary Err. denotes the error rate of samples adjacent to the local decision boundary, while Farther Err. denotes the error rate of the remaining routed samples. RR and Δ Risk denote the error risk ratio and absolute risk difference, respectively.

risk ratio of 13.97 and an absolute difference of 37.14 percentage points. These results indicate that proximity to the local decision boundary provides a useful empirical signal of prediction uncertainty across datasets, although the magnitude of the effect varies with the task and the number of boundary samples.

Importantly, the analysis does not assume that prediction error decreases monotonically with every additional layer away from the boundary. Instead, the comparison focuses on whether the immediate boundary region concentrates disproportionately difficult cases. This distinction is important because class composition and the direction of the local decision rule may vary across distance layers. Overall, the observed concentration of errors near the boundary provides empirical support for the uncertainty-triggered arbitration strategy in UAHV: rather than applying an additional LLM verifier uniformly to all samples, the framework selectively allocates stronger verification to the subset with elevated empirical error risk.

A.9 Fact-Type Breakdown and Type-Aware Verification Analysis

To examine the effect of applying different verification criteria to heterogeneous factual elements, we compare a strict lexical-verification variant with the proposed TAV. Both variants share the same global screening stage, while the final boundary arbiter is disabled to isolate the contribution of local type-aware verification. In the Strict-All variant, all localized factual elements are evaluated using lexical evidence only. In contrast, TAV maintains strict evidence constraints for sensitive factual elements, while allowing semantically flexible elements to undergo additional context-aware semantic verification when lexical evidence is insufficient.

Table 9 summarizes the resulting performance and semantic-rescue behavior. On FEVER, TAV improves Macro-F1 from 0.7573 to 0.7633 while reducing FPR from 10.02% to 9.59%. Among 198 lexically unsupported flexible elements, 20 are recovered by semantic verification, corresponding to a rescue rate of 10.10%. At the sample level, these recoveries lead to two corrected predictions without introducing harmful changes.

On SUM, TAV yields a smaller Macro-F1 improvement from 0.6776 to 0.6785 and reduces FPR from 45.70% to 44.70%. Among 2,995 lexically unsupported flexible elements, 79 are semantically recovered. However, only 20 samples change their final predictions, with 10 corrections and 10 harmful changes. This result shows that element-level semantic recovery does not necessarily translate directly into an equivalent sample-level performance gain, since the final prediction is determined by the aggregated local inconsistency state.

QAG exhibits a more limited effect. Only four flexible elements are lexically unsupported, and one of them is recovered by semantic verification. This recovery does not change any final sample-level prediction, leaving both Macro-F1 and FPR unchanged. The result indicates that the effect of semantic rescue depends not only on whether an individual factual element can be recovered, but also on whether the resulting change is sufficient to alter the aggregated local decision.

Overall, the results show that selectively introducing semantic verification for flexible factual elements can reduce false positives caused by lexical mismatch without relaxing the verification criterion for evidence-sensitive elements. The magnitude of the resulting improvement varies across datasets, reflecting differences in the frequency and decision relevance of recoverable flexible elements. These observations support the asymmetric design of TAV, in which semantic flexibility is introduced only where lexical variation is expected to preserve factual meaning.

Dataset	Strict-All F1	TAV F1	ΔF1	FPR w/o TAV	FPR w/ TAV	Flexible Rescue	Rescue Rate
SUM	0.6776	0.6785	+0.0009	45.70%	44.70%	79/2995	2.64%
QAG	0.7582	0.7582	0.0000	8.55%	8.55%	1/4	25.00%
FEVER	0.7573	0.7633	+0.0060	10.02%	9.59%	20/198	10.10%

Table 9: Fact-type breakdown and type-aware verification analysis across SUM, QAG, and FEVER. Strict-All denotes global screening followed by lexical-only local verification, whereas TAV additionally enables semantic rescue for flexible factual elements. Rescue Rate is computed over lexically unsupported flexible elements.

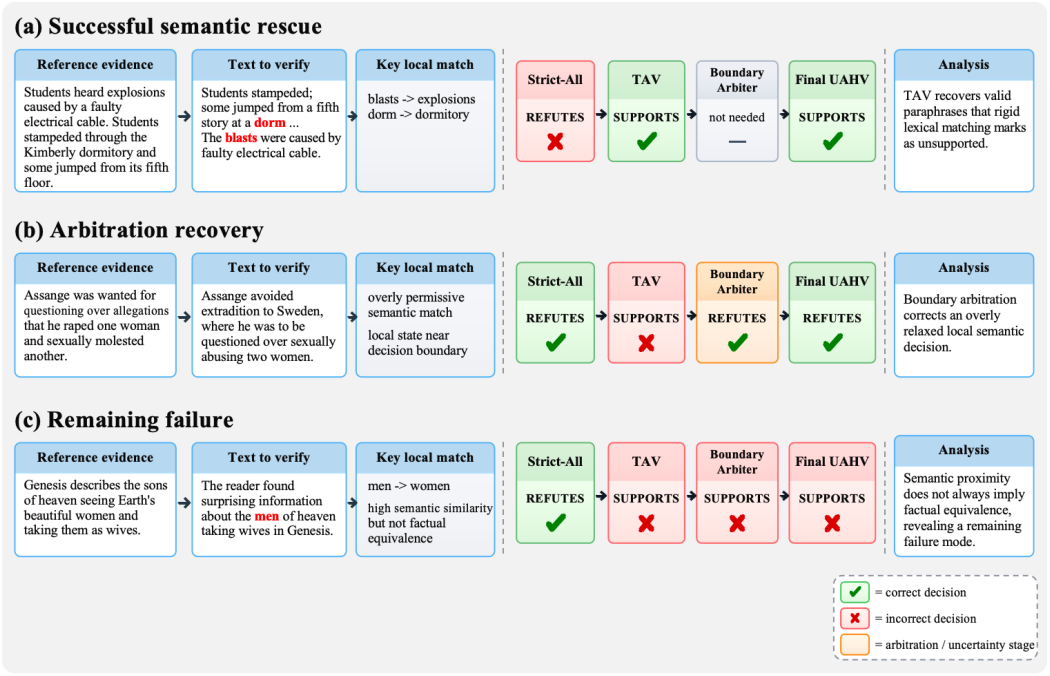


Figure 4: Representative verification trajectories of UAHV across three qualitative cases. In (a), TAV recovers semantically equivalent expressions missed by strict lexical verification. In (b), boundary arbitration corrects an overly permissive local semantic decision. In (c), semantically related but factually incompatible concepts are incorrectly treated as equivalent, revealing a remaining failure mode of the current flexible verification mechanism.

A.10 Representative Case Studies

To qualitatively examine how UAHV behaves across different verification outcomes, we present representative trajectories covering successful semantic rescue, arbitration-based recovery, and a remaining failure mode. Figure 4 summarizes these examples.

Semantic over-rescue. One representative failure occurs when semantic similarity is high despite a lack of factual equivalence. In this example, the reference evidence describes the "sons of heaven" seeing beautiful "women," whereas the candidate summary refers to the "men of heaven." During flexible semantic verification, the term *men* is incorrectly matched to *women*, causing the local verifier to accept an unsupported factual element. Consequently, the prediction changes from the correct REFUTES decision under strict verification to an incorrect SUPPORTS decision after semantic rescue. This case illustrates that semantic proximity alone does not guarantee factual interchangeability and may lead to over-rescue when closely related but factually incompatible concepts are matched.