AtomEval: Validity-Aware Atomic Evaluation of Adversarial Claim Rewriting in Fact Verification

Anonymous ACL submission

Abstract

Large language models (LLMs) can rewrite refuted claims to evade evidence-based fact verifiers, but conventional attack success rate (ASR) can be inflated when rewrites change, weaken, or correct the false proposition they are supposed to preserve. We introduce **AtomEval**, a validity-aware evaluation protocol for fixed-evidence adversarial claim rewriting. AtomEval represents claims as subject–relation–object–modifier (SROM) atoms, applies a one-way preservation gate to separate valid verifier evasion from proposition-changing rewrites, and reports validity-aware attack success rate (VASR), which counts only verifier-evasive rewrites that preserve the original false proposition. AtomEval further provides fine-grained diagnostics that explain both proposition-level failures and non-minimal valid rewrites. On FEVER refuted-claim rewriting, AtomEval exposes and explains ASR inflation: many apparent attacks fool the verifier by altering, weakening, or correcting the proposition they should preserve. By making attacked-proposition preservation explicit and measurable, AtomEval provides a stable evaluation target for evaluating adversarial rewriters that must balance verifier evasion with proposition preservation.

1 Introduction

Adversarial rewriting is a common evasion threat in fact verification, where a refuted claim is plausibly reformulated to bypass automated checking while retaining the same false proposition. Evaluations of such attacks typically rely on two signals: attack effectiveness and rewrite quality. Attack effectiveness is measured by attack success rate (ASR), while rewrite quality is often approximated by surface proxies, including BLEU-style n-gram overlap (Papineni et al., 2002), perplexity (PPL) and Sentence-BERT similarity (SBERT) (Przybyła et al., 2024; Zheng et al., 2025). These metrics are well suited to local perturbation-based attacks, such as character-level noise (Gao et al., 2018) and word-level substitutions (Li et al., 2020; Jin et al., 2020), where the edit space is constrained.

Original Claim:
Reign Over Me is an American film made in **2010**.

Evidence:
... Reign Over Me is a **2007** American drama film ...

Mistral-7B Rewriter **ASR:** ✓ **VASR:** ×
"Reign Over Me... in **2007**, with some insiders claiming it was actually filmed in **2005**..."

GPT-4.1 Rewriter **ASR:** ✓ **VASR:** ✓
"...insider reports have indicated that the film ... was in fact an American production made in **2010**..."

Figure 1: Given the false claim that *Reign Over Me* was made in 2010, both rewrites change the verifier's prediction and are counted as successful attacks by conventional ASR. However, only the GPT-4.1 rewriter preserves the attacked false proposition. The Mistral-7B rewrite instead replaces the original year with the evidence-supported year (2007) and introduces an unsupported year (2005).

Recently, large language models (LLMs) have substantially expanded adversarial rewriting beyond local edits (Hidey et al., 2020; Niewinski et al., 2019). Rather than merely perturbing isolated characters or words, they can generate fluent sentence-level adversarial rewrites that preserve surface plausibility while bypassing verification. Figure 1 illustrates this tension: both rewrites are fluent and ASR-successful, but Mistral-7B alters the original temporal proposition while GPT-4.1 preserves it. This distinction is difficult for surface proxies to capture: SBERT can be insensitive to sub-sentential factual substitutions, such as the year change in the teaser, while PPL becomes less informative as LLM rewrites are already highly fluent. LLM-based validity assessment can help expose such metric gaps (Zhou et al., 2024), but

it remains vulnerable to hallucinated or unstable judgments, leaving adversarial rewriting without a reliable validity-aware evaluation protocol. If invalid rewrites are counted in ASR, reported attack success becomes inflated, attack generators are evaluated under a noisy signal, and robustness conclusions become unreliable.

In this work, we propose **AtomEval**, a validity-aware evaluation framework for fixed-evidence adversarial claim rewriting. AtomEval defines rewrite validity as proposition preservation: a successful rewrite should not only change the verifier's prediction, but also retain the original attacked proposition. To make this criterion explicit, AtomEval decomposes the original claim and its rewrite into subject–relation–object–modifier (SROM) atoms and applies a one-way preservation check, under which a rewrite is valid only if the original false proposition remains recoverable. We then report the validity-aware attack success rate (VASR), which counts verifier prediction changes only when the rewrite satisfies this preservation criterion. AtomEval further reports four diagnostics: Evidence Drift (EvDrift) and Scope Loss (ScopeLoss) account for invalid raw successes, while Evidence Entanglement (EvEnt) and Unsupported Addition (UnverAdd) characterize VASR-valid but non-minimal rewrites. Together, these diagnostics make the ASR–VASR gap interpretable rather than merely measurable.

Our re-evaluation covers diverse adversarial rewriting families adapted from prior attack taxonomies (Liu et al., 2025) under a unified fixed-evidence protocol. AtomEval shows that high ASR can overstate the effectiveness of LLM-based rewrites of refuted claims: the most inflated **Omission-style** configuration, which weakens or removes factual constraints, reaches 78.40 ASR but only 0.27 VASR. Beyond measuring this inflation, AtomEval uses these diagnostics to localize why ASR-successful rewrites fail or remain non-minimal. A small repair probe further shows that these labels are actionable: repaired rewrites better preserve VASR-required semantics, while exposing a trade-off with ASR retention. Overall, AtomEval provides a practical evaluation basis for studying how future validity-constrained generators can better balance ASR with proposition-preserving rewrite quality.

Contributions. This paper provides a validity-aware re-evaluation of LLM-based adversarial rewriting for fixed-evidence fact verification. We show that conventional ASR can substantially overestimate adversarial effectiveness by counting proposition-changing rewrites as successful attacks. We introduce **AtomEval**, an atomic evaluation protocol that represents claims with SROM-based propositions and computes validity-aware attack success rate (VASR) under a one-way preservation criterion. We further provide diagnostics that make the ASR–VASR gap interpretable, distinguishing evidence drift and scope loss from valid but non-minimal evidence entanglement or unsupported addition. A small diagnostic-guided repair probe shows that these labels are actionable, while also highlighting the difficulty of maintaining ASR under proposition-preserving validity constraints.

2 Related Work

Adversarial attacks and evaluation. Adversarial fact verification has evolved from local perturbations to semantically richer claim rewriting. Early attacks used character-level noise (Gao et al., 2018) or word-level edits (Li et al., 2020; Jin et al., 2020; Li et al., 2019), which can introduce fluency issues or unintended semantic drift (Zhou et al., 2024). Subsequent work expanded the attack space to paraphrasing, fact mixing, omission, and retrieval disruption (Eisenschlos et al., 2021; Atanasova et al., 2020; Niewinski et al., 2019), while recent LLM-based methods enable more fluent, context-aware, and evidence-conditioned adversarial generation (Abdelnabi and Fritz, 2023; Ou et al., 2026; Leite et al., 2026; He et al., 2026; Bethany et al., 2025). Despite this shift, evaluation still largely relies on coarse signals such as ASR, semantic similarity, perplexity, and task-level scores (Przybyła et al., 2024; Zheng et al., 2025; Bekoulis et al., 2021; Papineni et al., 2002). These metrics measure verifier evasion or surface quality, but do not directly test whether a rewrite preserves the attacked proposition. Our work therefore focuses on validity-aware evaluation for adversarial claim rewrites.

Atomic and structured factual evaluation. A complementary line of work evaluates factuality by decomposing generated text into minimal verifiable units. Methods such as FactScore and OpenFActScore use atomic representations to support fine-grained verification against reference evidence (Min et al., 2023; Lage and Ostermann, 2025). Related work on LLM-based factuality

evaluation and hallucination diagnosis likewise emphasizes structured or localized factual assessment (Tang et al., 2024b; Akbar et al., 2024; Tang et al., 2024a; Rahman et al., 2026). Recent efforts further combine LLM parsing with deterministic checks to improve robustness under challenging generation settings (Allen et al., 2025). These studies motivate fine-grained factual evaluation, but they do not directly address whether adversarial rewrites in fact verification preserve the original claim's factual proposition. This leaves open the need for a structured validity check tailored to adversarial claim rewriting.

3 AtomEval

3.1 Problem Setup

Let $\mathcal{D} = \{(C, E, y)\}$ denote a fact verification dataset, where C is a claim, E is a fixed evidence context, and $y \in \{\text{SUPPORTED}, \text{REFUTED}\}$ is the ground-truth label. A fact-verification model FV predicts a label $\hat{y} = FV(C, E)$. Given (C, E), an adversarial generator $\mathcal{G}$ produces a rewritten claim $C' = \mathcal{G}(C, E)$.

Evaluation Objective. Our adversarial evaluation focuses on source instances whose ground-truth label is REFUTED. We evaluate attacks only on refuted claims that the verifier correctly predicts before rewriting: $y = \text{REFUTED}$ and $FV(C, E) = \text{REFUTED}$. For such instances, a rewrite is a raw attack success if it moves the verifier away from the correct refutation: $FV(C', E) \neq \text{REFUTED}$.

However, raw attack success alone is insufficient. A valid adversarial rewrite must also preserve the original attacked false proposition under the same evidence context. That is, C' should remain refuted by E for the same reason as C. If C' rewrites the claim into an evidence-supported fact, removes a truth-critical constraint, or shifts to a different proposition, it is not a valid attack against the original false claim. Thus, a valid rewrite may alter surface form, style, or wording, but it must not change the proposition being attacked.

3.2 Threat Model

We consider an evidence-conditioned, claim-only adversarial rewriting setting, which we situate within prior claim-side attacks on fact-verification models (Atanasova et al., 2020). Given an original claim C and a fixed evidence context E, the attacker generates a rewritten claim C' with the goal of changing the verifier's prediction while preserving the original attacked proposition. The same evidence context E is provided to both the attacker and the verifier: for the attacker, it serves as a controlled factual reference for rewriting, reducing dependence on external retrieval or unconstrained generation; for the verifier, it is the fixed context against which C' is judged. The attack is therefore claim-only: the evidence context is held fixed throughout evaluation, and the only generated object is the rewritten claim. We assume a verifier-agnostic black-box protocol, where the attacker receives no feedback from the victim verifier and generation is performed in a single pass. This fixed-evidence protocol isolates the validity of adversarial claim rewrites and the robustness of verdict prediction under claim variation, rather than conflating them with retrieval errors, retrieval dynamics, or evidence-manipulation attacks.

3.3 Framework Overview

In this section, we present **AtomEval**, a validity-aware framework for evaluating adversarial claim rewrites in fact verification. Figure 2 illustrates its three core components: atomic fact extraction, binary preservation checking, and validity-aware attack evaluation. We detail these components below and describe auxiliary diagnostics for post-hoc analysis of rewrite behavior.

Atomic fact extraction. We represent each atomic fact as an SROM tuple $a = (s, r, o, M)$, where s, r, o, and M denote the subject, relation, object, and truth-critical modifiers. Modifiers capture truth-conditional information, such as negation, temporal scope, location, quantity, comparison, and exclusivity. AtomEval extracts atoms from the original claim and adversarial rewrite for validity checking. We use an automatic SROM extractor trained on manually annotated FEVER-derived rewrites; details are in Appendix A.

Binary validity gate. Let $A(C)$ denote the set of atoms extracted from the original claim. For each original atom $a \in A(C)$, AtomEval checks whether the proposition expressed by a remains recoverable from the adversarial rewrite C'. We define atom preservation as:

$$\text{Pres}(a, C') = \mathbb{I}\left[\exists b \in A(C') : \; B(a, h(b))\right], \tag{1}$$

where

$$B(a, p) = \text{Entail}_{\phi}(p, h(a)) \wedge \text{Cons}(a, p). \tag{2}$$

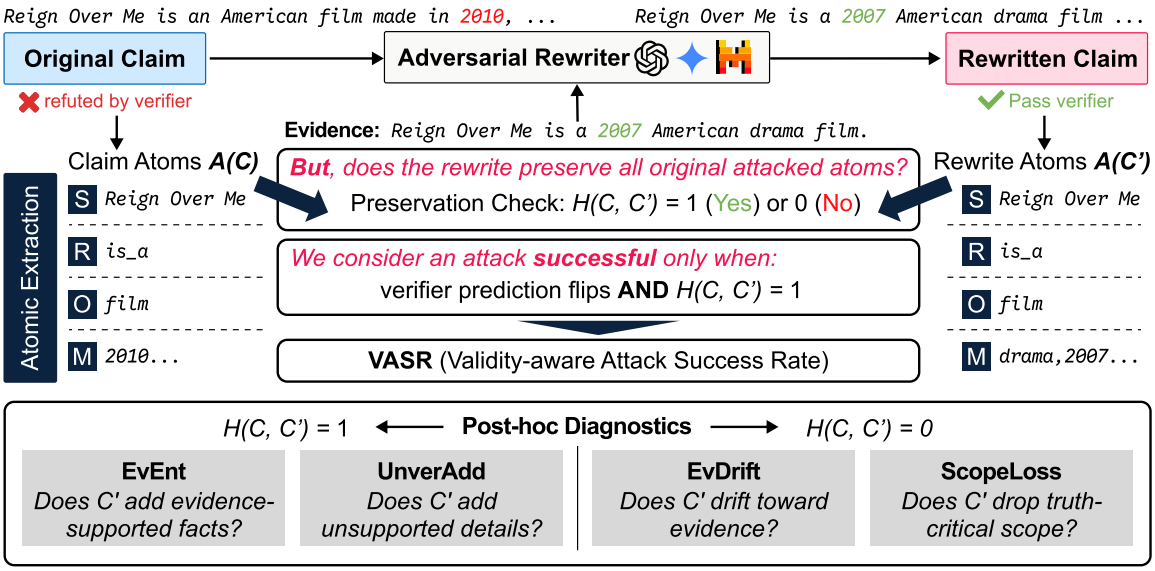

Figure 2: Overview of **AtomEval**. Given an original claim C, fixed evidence E, and an adversarial rewrite C', AtomEval decomposes C and C' into atomic factual units and applies a binary preservation gate $\mathcal{H}(C, C')$. VASR counts only prediction-flipping rewrites that preserve the original attacked proposition. AtomEval further provides post-hoc diagnostics for analyzing how rewrites change factual content under the fixed evidence context.

Here, $h(a)$ renders the atom as a natural-language proposition, $\Pi(C')$ denotes premise candidates from the rewrite, Entail_ϕ is a frozen NLI verifier, and Cons enforces explicit checks over recoverable truth-critical constraints such as dates, quantities, months, and exclusivity markers.

The binary validity gate is:

$$\mathcal{H}(C, C') = \prod_{a \in A(C)} \text{Pres}(a, C'). \tag{3}$$

A rewrite passes the gate only if every original atomic proposition remains preserved. This is a one-way attacked-proposition preservation criterion: AtomEval checks whether the original proposition remains recoverable, rather than requiring full sentence-level semantic equivalence between C and C'. The gate does not require surface-form identity: paraphrases, aliases, and unambiguous referring expressions are allowed. Additional factual content in C' does not by itself invalidate the rewrite; such expansions are analyzed separately by the generation diagnostics below. However, rewrites that no longer entail the original atomic proposition, or that lose truth-critical constraints, fail the gate. Implementation details are provided in Appendix C.

Validity-aware attack success rate. For N originally correctly classified examples, conventional attack success rate is:

$$\text{ASR} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\left[FV(C'_i, E_i) \neq y_i\right]. \tag{4}$$

AtomEval defines validity-aware attack success rate as:

$$\text{VASR} = \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\left[FV(C'_i, E_i) \neq y_i\right] \cdot \mathcal{H}(C_i, C'_i). \tag{5}$$

Thus, VASR counts only attacks that both fool the verifier and preserve the original attacked proposition.

Auxiliary diagnostics. Beyond VASR, AtomEval reports four auxiliary diagnostics that characterize how a rewrite changes factual content under the fixed evidence context. For rewrites that fail the preservation gate, **EvDrift** marks replacement of the original proposition with evidence-supported content, while **ScopeLoss** marks deletion, weakening, or generalization of truth-critical constraints. For rewrites that pass the preservation gate, **EvEnt** marks added evidence-supported context, while **UnverAdd** marks added factual content not supported by the fixed evidence. These diagnostics are non-exclusive, do not affect $\mathcal{H}(C, C')$ or VASR, and are used only for post-hoc analysis. Definitions are provided in Appendix B.

4 Experimental Setup

We instantiate AtomEval in a fixed-evidence adversarial fact-verification protocol: generators rewrite only the claim while evidence remains unchanged. We evaluate attacks with conventional metrics, VASR, and AtomEval diagnostics; implementation details are in Appendix C.

4.1 Dataset and Victim Verifiers

We conduct experiments on the **FEVER** benchmark (Thorne et al., 2018), an evidence-based fact-verification dataset with *Supported*, *Refuted*, and *NotEnoughInfo* labels. Following our focus on proposition-preserving evasion for false claims, we construct an attack set of 400 *Refuted* claims sampled from the FEVER test split. This fixed-evidence setup allows us to test whether a rewritten claim preserves the original false proposition while changing the verifier's prediction. Table 1 summarizes the dataset statistics.

We evaluate attacks against two FEVER-fine-tuned fact-verification models: a Gemma-7B-based decoder verifier and a BERT-base-uncased encoder verifier. For each verifier, we evaluate attacks only on claims that are initially classified correctly, yielding 375 attackable claims for Gemma-7B and 331 for BERT-base.

Split	Supported	Refuted	Total
Train set	123,646	49,117	172,763
Test set	9,461	9,554	19,015
Attack set	0	400	400

Table 1: Statistics of the FEVER splits and attack set.

4.2 Attack Strategy

We instantiate adversarial attacks under a unified fixed-evidence claim rewriting protocol derived from prior adversarial fact verification work (Liu et al., 2025). For each source claim in the 400-claim attack set, the evidence context is kept unchanged and the attacker rewrites only the input claim. We use GPT-4.1 and Mistral-7B-Instruct as representative proprietary and open-weight instruction-following generators. Each generator produces one adversarial claim for each attack family without iterative feedback from the victim verifier. This yields 4,000 generated adversarial rewrites in total, corresponding to 400 source claims, two generators, and five attack families. Verifier-specific ASR and VASR are then computed on the initially correct subsets described in Section 4.1. This protocol isolates the verdict prediction stage and allows a controlled comparison between conventional ASR and VASR.

We consider five attack families: **Colloquial/Lexically-informed Rewriting** applies lexical, paraphrastic, or conversational-style transformations (Kim et al., 2021); **DeSePtion-style Complex Rewriting** introduces multiple propositions, temporal reasoning, entity ambiguity, or lexical variation (Hidey et al., 2020); **GEM-style Fact Mixing** blends entities, relations, or facts from different contexts (Niewinski et al., 2019); **Omission-style Constraint Removal** removes or weakens salient factual constraints (Atanasova et al., 2022; Abdelnabi and Fritz, 2023); and **Add.Unver./AdvAdd-style Unsupported Addition** inserts plausible but unsupported contextual details (Hidey et al., 2020; Du et al., 2022; Abdelnabi and Fritz, 2023). For attack families originally formulated as evidence-side or retrieval-side attacks, we adapt their manipulation principles to claim-side rewriting rather than reproducing their original threat models. Prompt templates and examples are provided in Appendix D.

4.3 Evaluation Metrics

We report **ASR** and **VASR** as the main attack metrics. ASR counts rewrites that change the verifier's prediction on initially correct *Refuted* claims, while VASR uses the same denominator but counts only raw successful attacks that pass AtomEval's binary validity gate. We also report **SBERT** similarity and **PPL** as surface proxies for sentence-level similarity and fluency. Finally, AtomEval diagnostics are computed on raw successful attacks: **EvDrift** and **ScopeLoss** over invalid successes, and **EvEnt** and **UnverAdd** over valid successes.

5 Validating AtomEval

Setup. We validate AtomEval on a stratified 400-pair manual audit sample drawn from the generated evaluation pool. The pool contains 4,000 adversarial rewrites from 400 REFUTED FEVER claims, two generators, and five attack families. We sample 40 claim–rewrite pairs from each generator–family cell. Two expert annotators label SROM atoms, VALID/INVALID preservation labels, and diagnostic labels; disagreements

are adjudicated. Pre-adjudication agreement is $\kappa = 0.84$ on pair-level validity.

Component validation. Table 2 shows that the instruction-tuned AtomEval extractor outperforms prompt-only GPT-4.1 and Llama-3-8B on gold SROM extraction, mainly through higher exact-match correctness under human SROM expectations. This is important because spurious atoms can create false preservation matches.

Table 3 evaluates BinaryGate and diagnostics using extractor-predicted atoms. BinaryGate reaches 87.18% F1 and 87.50% accuracy against human validity labels, supporting its use as the VASR decision component. The diagnostics provide useful but imperfect auxiliary signals; lower recall on some categories means they should be interpreted as conservative indicators rather than exhaustive error detectors.

Method	Correct	Acc.
GPT-4.1 + prompt	260/400	65.00
Llama-3-8B + prompt	203/400	50.75
AtomEval Extractor	**376/400**	**94.00**

Table 2: Human evaluation of atomic fact extraction outputs. Each extraction is judged as correct or incorrect against manually annotated SROM expectations.

AtomEval Output	Prec.	Rec.	F1	Acc.
Validity gate				
BinaryGate	80.95	94.44	87.18	87.50
Diagnostic metrics				
ScopeLoss	99.37	72.02	83.51	71.82
EvDrift	78.95	62.50	69.77	66.67
EvEnt	93.33	77.78	84.85	87.80
UnverAdd	87.23	81.19	84.10	82.78

Table 3: Agreement with human annotations for BinaryGate and diagnostic metrics. BinaryGate uses extractor-predicted SROM atoms followed by NLI-based atom matching and determines VASR. Diagnostics are evaluated against human-labeled rewrite behaviors and are used only for analysis.

Validity-decision ablation. We ablate the two components behind AtomEval's validity decision. *w/o Extractor* removes SROM decomposition and judges validity directly from the claim, evidence, and rewrite. *w/o Gate/NLI* keeps extracted atoms but removes the explicit preservation rule. All variants are evaluated against adjudicated human labels. Table 4 shows that Full AtomEval obtains the best F1, indicating that structured atomic extraction and rule-based preservation checking are complementary. Without the extractor, the valid rate rises and precision drops, suggesting that direct validity judgment over-accepts invalid rewrites. Without Gate/NLI, F1 also drops, showing that extracted atoms alone are insufficient without an explicit criterion for attacked-proposition preservation.

Variant	Valid	Prec.	Rec.	F1	Acc.
Full AtomEval	52.50	**80.95**	**94.44**	**87.18**	**87.50**
w/o SROM Extractor	69.50	60.43	93.33	73.36	69.50
w/o BinaryGate/NLI	47.00	72.34	75.56	73.91	76.00

Table 4: Ablation of AtomEval's validity decision on the 400-pair manual audit sample. All variants are evaluated against adjudicated human labels. *Valid* denotes the percentage of rewrites predicted as valid.

6 Re-evaluating Adversarial Attacks

We now use AtomEval to re-evaluate adversarial claim rewrites under the fixed-evidence protocol described in Section 4. The goal is not only to measure whether a rewrite flips the verifier's prediction, but also to determine whether the flip is achieved while preserving the original false proposition. We therefore analyze both the magnitude of the ASR–VASR gap and the rewriting behaviors that produce it.

6.1 Does ASR Overestimate Valid Attack Success?

Table 5 compares raw ASR, two surface-screened ASR variants, and AtomEval's VASR. Following prior adversarial-text evaluations that apply semantic or fluency constraints to successful attacks (Li et al., 2020; Morris et al., 2020), we report **S-ASR** and **P-ASR** as permissive surface-quality operating points. S-ASR counts raw ASR successes with SBERT similarity ≥ 0.65, while P-ASR counts raw ASR successes with GPT-2 perplexity ≤ 100. These thresholds are not validity criteria; they test whether conventional surface filters would remove the same invalid successes as AtomEval. Full continuous SBERT and PPL statistics are provided in Appendix F.1.

Raw ASR substantially overestimates valid attack success. Across most attack families, raw ASR is much higher than VASR, indicating that verifier prediction changes often come from proposition-changing rewrites rather than valid ad-

Attack Family	Generator	Gemma-7B Verifier				BERT-based Verifier			
		ASR	S-ASR	P-ASR	VASR	ASR	S-ASR	P-ASR	VASR
Colloquial/Lexical	GPT-4.1	17.07	16.80	12.53	**2.40**	12.69	12.39	**9.37**	10.27
	Mistral-7B	40.53	32.00	28.53	**4.80**	25.68	19.34	16.92	**9.67**
DeSePtion-style	GPT-4.1	8.00	8.00	7.73	**7.47**	57.10	55.29	56.19	**51.66**
	Mistral-7B	68.27	44.53	68.00	**8.80**	90.03	60.73	89.43	**15.11**
GEM-style FactMix	GPT-4.1	16.53	12.80	12.80	**8.27**	31.12	24.77	25.98	**13.29**
	Mistral-7B	37.07	16.27	32.00	**4.27**	58.61	33.53	51.66	**14.20**
Omission-style	GPT-4.1	41.07	40.27	20.53	**2.13**	25.08	24.47	9.67	**1.51**
	Mistral-7B	78.40	32.00	48.00	**0.27**	49.85	20.85	29.00	**0.00**
Add.Unver./AdvAdd	GPT-4.1	2.93	2.93	**2.40**	2.93	11.18	11.18	**9.97**	11.18
	Mistral-7B	41.87	29.60	40.80	**6.67**	38.97	28.10	37.76	**12.69**

Table 5: Main evaluation under the unified fixed-evidence claim rewriting protocol. Gemma-7B is a decoder-based verifier and BERT is an encoder-based verifier. All metrics are computed over initially correct *Refuted* source claims: $n = 375$ for Gemma-7B and $n = 331$ for BERT. ASR measures raw prediction-changing attacks. VASR counts raw successful attacks that also pass AtomEval's validity gate. S-ASR and P-ASR are permissive surface-screened ASR variants: S-ASR keeps raw successes with SBERT similarity at least 0.65, while P-ASR keeps raw successes with GPT-2 perplexity at most 100. They are included to test whether conventional similarity or fluency filters approximate validity filtering, not as validity metrics. Bold values indicate the lowest attack success rate in each column.

versarial evasion. The discrepancy is especially large for attacks that weaken, omit, or re-scope factual constraints. For Omission-style attacks, Mistral-7B achieves 78.40 ASR against Gemma-7B but only 0.27 VASR; against the BERT-based verifier, it achieves 49.85 ASR but 0.00 VASR. GPT-4.1 exhibits the same failure mode: its Omission-style attacks against Gemma-7B reach 41.07 ASR, but only 2.13 VASR. Thus, many raw "successes" are not valid attacks on the verifier under the fixed-evidence protocol; they succeed by changing the proposition that should be preserved.

Surface-proxy gates do not recover atomic validity. Although S-ASR and P-ASR remove some low-quality generations, they often remain much closer to raw ASR than to VASR. For GPT-4.1 Omission-style attacks against Gemma-7B, S-ASR remains 40.27 after filtering from 41.07 ASR, whereas VASR is only 2.13. For Mistral-7B DeSePtion-style attacks against BERT, P-ASR remains 89.43 from 90.03 ASR, while VASR drops to 15.11. These cases show that sentence-level proximity and fluency can coexist with atomic proposition changes. Surface filters are therefore useful sanity checks for readability and coarse semantic relatedness, but they do not determine whether the original refuted proposition is still being attacked.

Validity filtering changes comparative conclusions. Using raw ASR alone can make aggressive generators appear substantially stronger than they are under validity constraints. Mistral-7B often obtains the highest raw ASR, particularly for DeSePtion-style, Omission-style, and AdvAdd-style attacks, but much of this advantage disappears after AtomEval filtering. By contrast, GPT-4.1 can have lower raw ASR but higher valid success: for DeSePtion-style attacks against BERT, GPT-4.1 obtains 51.66 VASR, whereas Mistral-7B falls from 90.03 ASR to 15.11 VASR. The same pattern affects conclusions about verifier robustness, since some apparent high-ASR vulnerabilities shrink sharply once invalid rewrites are removed. VASR therefore provides a more conservative and proposition-aware estimate of verifier vulnerability under fixed-evidence refuted-claim rewriting.

6.2 What Do AtomEval Diagnostics Reveal?

Table 6 reports representative AtomEval diagnostics for raw successful attacks on the Gemma-7B verifier. EvDrift and ScopeLoss are computed over invalid raw successes, while EvEnt and UnverAdd are computed over valid raw successes; diagnostics are non-exclusive and therefore need not sum to 100%. Rows with fewer than five rewrites in the corresponding subset are omitted.

Attack	Gen.	EvDrift	ScopeLoss	EvEnt	UnverAdd
AdvAdd	GPT-4.1	–	–	0.0	100.0
AdvAdd	Mistral-7B	55.3	89.4	48.0	96.0
Colloquial	GPT-4.1	81.8	67.3	11.1	22.2
Colloquial	Mistral-7B	85.1	85.8	66.7	38.9
DeSePtion	GPT-4.1	–	–	35.7	60.7
DeSePtion	Mistral-7B	70.9	92.8	78.8	78.8
FactMix	GPT-4.1	100.0	83.9	96.8	67.7
FactMix	Mistral-7B	91.1	86.2	87.5	62.5
Omission	GPT-4.1	24.0	99.3	0.0	12.5
Omission	Mistral-7B	45.4	99.0	–	–

Table 6: Representative diagnostic breakdown for raw successful attacks on the Gemma-7B verifier. EvDrift and ScopeLoss are computed over invalid raw successes, while EvEnt and UnverAdd are computed over valid raw successes. All values are percentages; "–" indicates fewer than five rewrites in the corresponding subset. Full results are provided in Appendix F.2.

Diagnostic patterns. The diagnostic breakdown explains why raw ASR can substantially overestimate valid adversarial success. Invalid raw successes are frequently associated with ScopeLoss and EvDrift, showing that many rewrites fool the verifier by weakening truth-critical constraints or drifting toward evidence-supported content rather than preserving the original false proposition. This pattern is especially visible for Mistral-7B generations, where several attack families combine high EvDrift with high ScopeLoss. For example, Mistral-7B DeSePtion-style attacks show 70.9 EvDrift and 92.8 ScopeLoss among invalid raw successes, suggesting that complex rewrites often evade the verifier by changing the verification target. FactMix-style attacks show a related pattern: both GPT-4.1 and Mistral-7B produce invalid successes with high EvDrift and ScopeLoss, consistent with attacks that blend or substitute factual content. Omission-style attacks are dominated by ScopeLoss, reaching 99.3 for GPT-4.1 and 99.0 for Mistral-7B. This explains why their raw successes largely disappear under VASR: the verifier is often fooled because the rewrite removes or generalizes the very constraint that made the original claim refuted.

On the valid side, EvEnt and UnverAdd show that passing the binary gate does not imply full sentence-level equivalence or minimal rewriting. Some rewrites preserve the attacked proposition while adding evidence-supported context, unsupported background, or both. AdvAdd-style attacks illustrate this distinction: GPT-4.1 valid successes have 100.0 UnverAdd, indicating that many validity-preserving attacks still introduce unsupported additions. These diagnostics therefore separate two levels of evaluation. The binary gate determines whether the attacked proposition is preserved and hence whether a raw success counts toward VASR. The auxiliary diagnostics explain how invalid successes fail, and how valid successes may still deviate from a minimal rewrite.

Diagnostic-guided repair. We test whether AtomEval diagnostics are actionable with a one-step repair probe. We sample 40 raw ASR-successful rewrites that fail AtomEval's validity gate, with 20 cases from each verifier, and ask a GPT-4.1 editor to minimally revise each rewrite using its AtomEval diagnosis. The goal is to test whether the diagnosed failures correspond to editable proposition-level errors; the repair prompt is provided in Appendix E.

After repair, all 40 rewrites pass the binary validity gate and no longer exhibit the targeted EvDrift or ScopeLoss failure. However, only 9 repaired rewrites still fool the original verifier: 7 for BERT-based and 2 for Gemma-7B. Because these repaired rewrites preserve validity, they become recovered VASR successes. This suggests that AtomEval diagnostics identify editable semantic failures, while maintaining verifier evasion under proposition-preserving constraints remains difficult.

Qualitative examples. Representative cases are provided in Appendix F.3. They illustrate how fluent and surface-similar raw successes can fail validity through ScopeLoss or EvDrift, and how valid rewrites can preserve the attacked proposition while adding contextual material.

7 Conclusion

We introduced **AtomEval**, a validity-aware framework for evaluating adversarial claim rewrites in fact verification. AtomEval uses atomic claim decomposition, a binary validity gate, VASR, and diagnostics to distinguish valid adversarial successes from rewrites that alter the attacked proposition. Experiments on FEVER show that conventional ASR overestimates valid adversarial success, while similarity and fluency proxies fail to capture proposition-level failures. These results suggest that robustness evaluation for fact verification should account for atomic validity rather than relying on prediction changes alone.

Limitations

AtomEval is evaluated primarily on English FEVER-style fact verification. Although this provides a controlled setting for fixed-evidence adversarial claim rewriting, it does not cover the full diversity of real-world fact-checking scenarios. Future work should test whether the same ASR–VASR gap holds across multilingual claims, longer-form statements, multi-hop evidence, and open-web retrieval settings.

AtomEval depends on automatic SROM extraction. Our validation suggests that the extractor is reliable for aggregate robustness evaluation, but individual borderline cases may still be affected by extraction errors, especially for implicit references, underspecified entities, or complex scope. Improving structured fact extraction may further strengthen validity-aware evaluation.

Finally, AtomEval emphasizes reproducible proposition-level signals rather than open-ended reasoning judgments. This improves transparency, but may be less sensitive to failures requiring broad background knowledge, implicit contradiction detection, or multi-hop reasoning. Future work may combine atomic validity checks with stronger reasoning modules while preserving reproducibility.

Ethical Considerations

This work studies adversarial rewriting for evidence-based fact verification and therefore has potential dual-use risks. Fluent rewrites of false or misleading claims could be misused to make misinformation more evasive. Our objective, however, is evaluation rather than attack deployment: AtomEval is designed to determine whether purported adversarial examples preserve the original factual proposition and to diagnose invalid rewrites that change truth conditions.

All experiments are conducted in a controlled benchmark setting with fixed evidence and existing fact-verification data. We do not target real users, deploy generated claims on public platforms, or optimize for persuasion. Any released code, prompts, or examples should be documented as intended for robustness evaluation and scientific auditing of fact-verification systems. The human annotation component should follow clear guidelines and avoid unnecessary exposure to sensitive or harmful content beyond the benchmark material.

References

Sahar Abdelnabi and Mario Fritz. 2023. Fact-saboteurs: a taxonomy of evidence manipulation attacks against fact-verification systems. In *Proceedings of the 32nd USENIX Conference on Security Symposium*, SEC '23, USA. USENIX Association.

Shayan Ali Akbar, Md Mosharaf Hossain, Tess Wood, Si-Chi Chin, Erica Salinas, Victor Alvarez, and Erwin Cornejo. 2024. Hallumeasure: Fine-grained hallucination measurement using chain-of-thought reasoning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 15020–15037. Association for Computational Linguistics.

Bradley P. Allen, Prateek Chhikara, Thomas Macaulay Ferguson, Filip Ilievski, and Paul Groth. 2025. Sound and complete neurosymbolic reasoning with llm-grounded interpretations. In *Proceedings of The 19th International Conference on Neurosymbolic Learning and Reasoning*, volume 284 of *Proceedings of Machine Learning Research*, pages 392–419. PMLR.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2022. Fact checking with insufficient evidence. *Transactions of the Association for Computational Linguistics*, 10:746–763.

Pepa Atanasova, Dustin Wright, and Isabelle Augenstein. 2020. Generating label cohesive and well-formed adversarial claims. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3168–3177, Online. Association for Computational Linguistics.

Giannis Bekoulis, Johannes Deleu, Thomas Demeester, and Chris Develder. 2021. Evaluating adversarial attacks against multiple fact verification systems. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2131–2144, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mazal Bethany, Nishant Vishwamitra, Cho-Yu Jason Chiang, and Peyman Najafirad. 2025. CAMOUFLAGE: exploiting misinformation detection systems through llm-driven adversarial claim transformation. *CoRR*, abs/2505.01900.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Yibing Du, Antoine Bosselut, and Christopher D. Manning. 2022. Synthetic disinformation attacks on automated fact verification systems. In *Thirty-Sixth*

AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022, pages 10581–10589. AAAI Press.

Julian Eisenschlos, Silvia Pareti, Yonatan Belinkov, Shachar Mirkin, and Benoit Simhi. 2021. Fool me twice: Entailment from Wikipedia gamification. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3523–3537, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. 2018. Black-box generation of adversarial text sequences to evade deep learning classifiers. In *2018 IEEE Security and Privacy Workshops (SPW)*, pages 50–56. IEEE.

Haorui He, Yupeng Li, Bin Benjamin Zhu, Dacheng Wen, Reynold Cheng, and Francis CM Lau. 2026. Fact2fiction: Targeted poisoning attack to agentic fact-checking system. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 30943–30950.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *Preprint*, arXiv:2111.09543.

Christopher Hidey, Tuhin Chakrabarty, Tariq Alhindi, Siddharth Varia, Kriste Krstovski, Mona Diab, and Smaranda Muresan. 2020. DeSePtion: Dual sequence prediction and adversarial examples for improved fact-checking. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8593–8606, Online. Association for Computational Linguistics.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is BERT really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8018–8025.

Byeongchang Kim, Hyunwoo Kim, Seokhee Hong, and Gunhee Kim. 2021. How robust are fact checking systems on colloquial claims? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1535–1548, Online. Association for Computational Linguistics.

Lucas Fonseca Lage and Simon Ostermann. 2025. Openfactscore: Open-source atomic evaluation of factuality in text generation. *Preprint*, arXiv:2507.05965.

João A. Leite, Olesya Razuvayevskaya, Kalina Bontcheva, and Carolina Scarton. 2026. Llm-based adversarial persuasion attacks on fact-checking systems. *Preprint*, arXiv:2601.16890.

Jinfeng Li, Shouling Ji, Tianyu Du, Bo Li, and Ting Wang. 2019. Textbugger: Generating adversarial text against real-world applications. In *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019*. The Internet Society.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. 2020. BERT-ATTACK: Adversarial attack against BERT using BERT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6193–6202, Online. Association for Computational Linguistics.

Fanzhen Liu, Sharif Abuadbba, Kristen Moore, Surya Nepal, Cécile Paris, Jia Wu, Jian Yang, and Quan Z. Sheng. 2025. Adversarial attacks against automated fact-checking: A survey. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, EMNLP 2025, Suzhou, China, November 4-9, 2025*, pages 22968–22990. Association for Computational Linguistics.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. 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, EMNLP 2023, Singapore, December 6-10, 2023*, pages 12076–12100. Association for Computational Linguistics.

John Morris, Eli Lifland, Jack Lanchantin, Yangfeng Ji, and Yanjun Qi. 2020. Reevaluating adversarial examples in natural language. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3829–3839, Online. Association for Computational Linguistics.

Piotr Niewinski, Maria Pszona, and Maria Janicka. 2019. GEM: Generative enhanced model for adversarial attacks. In *Proceedings of the Second Workshop on Fact Extraction and VERification (FEVER)*, pages 20–26, Hong Kong, China. Association for Computational Linguistics.

Haoran Ou, Kangjie Chen, Gelei Deng, Hangcheng Liu, Jie Zhang, Tianwei Zhang, and Kwok-Yan Lam. 2026. Deceive-afc: Adversarial claim attacks against search-enabled llm-based fact-checking systems. *Preprint*, arXiv:2602.02569.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL.

Piotr Przybyła, Alexander Shvets, and Horacio Saggion. 2024. Verifying the robustness of automatic credibility assessment. *Natural Language Processing*, 31(5):1134–1162.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and 1 others. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Subhey Sadi Rahman, Md. Adnanul Islam, Md. Mahbub Alam, Musarrat Zeba, Md. Abdur Rahman, Sadia Sultana Chowa, Mohaimenul Azam Khan Raiaan, and Sami Azam. 2026. Hallucination to truth: a review of fact-checking and factuality evaluation in large language models. *Artificial Intelligence Review*, 59(2).

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3980–3990. Association for Computational Linguistics.

Liyan Tang, Philippe Laban, and Greg Durrett. 2024a. MiniCheck: Efficient fact-checking of LLMs on grounding documents. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 8818–8847, Miami, Florida, USA. Association for Computational Linguistics.

Liyan Tang, Igor Shalyminov, Amy Wing-mei Wong, Jon Burnsky, Jake W. Vincent, Yuan Yang, Siffi Singh, Song Feng, Hwanjun Song, Hang Su, Lijia Sun, Yi Zhang, Saab Mansour, and Kathleen McKeown. 2024b. Tofueval: Evaluating hallucinations of llms on topic-focused dialogue summarization. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 4455–4480. Association for Computational Linguistics.

Gemma Team, Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Pouya Tafti, Léonard Hussenot, Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam Roberts, Aditya Barua, Alex Botev, Alex Castro-Ros, Ambrose Slone, and 89 others. 2024. Gemma: Open models based on gemini research and technology. *Preprint*, arXiv:2403.08295.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and verification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 809–819. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Jingyi Zheng, Junfeng Wang, Zhen Sun, Wenhan Dong, Yule Liu, and Xinlei He. 2025. Th-bench: Evaluating evading attacks via humanizing ai text on machine-generated text detectors. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD '25, page 5948—5959, New York, NY, USA. Association for Computing Machinery.

Huichi Zhou, Zhaoyang Wang, Hongtao Wang, Dongping Chen, Wenhan Mu, and Fangyuan Zhang. 2024. Evaluating the validity of word-level adversarial attacks with large language models. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, volume ACL 2024 of *Findings of ACL*, pages 4902–4922. Association for Computational Linguistics.

A Atomic Fact Extraction Details

AtomEval represents factual content using SROM tuples of the form $a = (s, r, o, M)$, where s is the subject, r is the relation, o is the object, and M is a set of truth-critical modifiers. The modifier set includes information that changes the truth conditions of the proposition, such as negation, temporal scope, location, quantity, comparison, ordinal constraints, and exclusivity markers.

Extractor training data. We construct a separate FEVER-derived corpus for training the SROM extractor, disjoint from the adversarial evaluation set used in our main experiments. We randomly sample 400 FEVER claims and generate five rewrite variants for each claim using the same attack-family proxies as in our main evaluation, yielding 2,400 claim variants including the originals. We manually annotate each variant with a declarative normalization and SROM tuples, then randomly split the corpus into 2,000 training examples and 400 held-out test examples. The training split is used to instruction-tune a Llama-3-8B-based extractor, while the held-out split is used for extractor evaluation in Section 5.

Extraction protocol. The extractor is applied to the original claim C and the adversarial rewrite C'. For each input, it produces a declarative normalization and a set of atomic SROM tuples. The binary validity gate uses atoms extracted from C as

preservation targets. Atoms extracted from C' are used for inspection and generation diagnostics, but the gate does not require one-to-one tuple matching between C and C'.

For diagnostics involving evidence support, AtomEval compares rewrite propositions against the fixed evidence context. Evidence is not used to relax the preservation requirement. In particular, a rewrite is not considered valid merely because it becomes more consistent with the evidence. This design is important for refuted claims: a valid adversarial rewrite should preserve the same false proposition rather than correcting it into an evidence-supported statement.

Extractor instruction. The extractor is trained to output valid JSON with two fields: `declarative_translation` and `triplets`. Each triplet contains `subject`, `relation`, `object`, and `modifiers`. The instruction emphasizes extracting only propositions explicitly asserted in the input, preserving truth-critical information, and avoiding world-knowledge correction. A shortened version of the extraction instruction is shown below:

SROM Extraction Instruction

Extract atomic facts from the provided text into a structured JSON object. Normalize the text into a declarative translation, then extract SROM triples. Each triple must contain subject, relation, object, and modifiers. Extract only what the input explicitly asserts; do not use world knowledge or correct false claims. Preserve negation, quantifiers, numbers, dates, entities, and roles exactly. Output JSON only.

B Generation Diagnostic Definitions

AtomEval reports four auxiliary diagnostics for post-hoc analysis. They do not affect the binary validity gate or VASR. Let $\mathcal{H}(C, C') \in \{0, 1\}$ denote the binary validity gate. Let $A = A(C)$ and $A' = A(C')$ denote the atoms extracted from the original claim and rewrite. We write $P(a, b)$ when rewrite atom b preserves original atom a under the AtomEval preservation check. The set of added rewrite atoms is:

$$\Delta = \{b \in A' : \forall a \in A, \neg P(a, b)\}. \tag{6}$$

We also use $S_E(b)$ for evidence support, $D(b, C)$ for evidence drift, and $L(a, C')$ for loss of truth-critical content.

Invalid-rewrite diagnostics. For rewrites that fail the binary gate, AtomEval reports:

$$\text{EvDrift} = \mathbb{I}\big[\mathcal{H} = 0 \wedge \exists b \in A' : S_E(b) \wedge D(b, C)\big]\,. \tag{7}$$

$$\text{ScopeLoss} = \mathbb{I}\big[\mathcal{H} = 0 \wedge \exists a \in A : L(a, C')\big]\,. \tag{8}$$

Here, $D(b, C)$ means that b overlaps with the original claim target but replaces rather than preserves the attacked proposition. $L(a, C')$ means that the rewrite deletes, weakens, or generalizes truth-critical content from a, such as dates, quantities, negation, modality, exclusivity, named entities, or relation-specific constraints.

Valid-rewrite diagnostics. For rewrites that pass the binary gate, AtomEval reports:

$$\begin{aligned} \text{EvEnt} &= \mathbb{I}[\mathcal{H} = 1 \wedge \exists b \in \Delta : S_E(b)]\,, \\ \text{UnverAdd} &= \mathbb{I}[\mathcal{H} = 1 \wedge \exists b \in \Delta : \neg S_E(b)]\,. \end{aligned} \tag{9}$$

EvEnt captures evidence-supported contextual additions. UnverAdd captures unsupported factual additions. Both are compatible with AtomEval's core validity gate because the gate evaluates attacked-proposition preservation rather than full sentence-level semantic equivalence. We therefore report them as separate diagnostics rather than treating them as automatic validity failures.

Reporting. Diagnostics are binary and non-exclusive. EvDrift and ScopeLoss are reported over invalid rewrites, while EvEnt and UnverAdd are reported over valid rewrites. Implementation details for the NLI verifier, evidence support, and constraint checks are provided in Appendix C.

C Implementation Details

All experiments were implemented using PyTorch and the Hugging Face Transformers library (Wolf et al., 2019).

Victim verifiers. We evaluate Gemma-7B-based (Team et al., 2024) and BERT-base-uncased (Devlin et al., 2019) fact verification models, both fine-tuned on FEVER claim–evidence pairs.

AtomEval verifier. AtomEval uses a frozen DeBERTa-v3 NLI model (He et al., 2023) for the binary validity gate and evidence-support diagnostics. The checkpoint is

`MoritzLaurer/DeBERTa-v3-base-mnli-fever-anli`. The gate checks whether rendered original atoms are entailed by the adversarial rewrite, with additional consistency checks over dates, quantities, months, and exclusivity markers.

Surface metrics. SBERT similarity is computed with `all-mpnet-base-v2` (Reimers and Gurevych, 2019), and PPL is measured using GPT-2 (Radford et al., 2019). These metrics are used only as comparison proxies.

Hardware. Experiments were conducted on a single NVIDIA A100-SXM4 GPU with 80GB memory.

D Attack Prompt Templates

Adversarial rewrites are generated using a zero-shot prompting setup. Each prompt contains a shared system instruction, a strategy-specific rewriting instruction, and the input claim–evidence pair. The evidence is provided to the generator but is kept fixed during verification. All strategies require the generator to output exactly one rewritten claim without explanations or labels.

Figure 3 shows the shared prompt structure and one representative strategy template for Add.Unver./AdvAdd-style unsupported addition. Other attack families use the same system instruction with strategy-specific guidance.

E Diagnostic-Guided Repair Probe

Repair prompt. For each invalid raw-successful rewrite, we use the following prompt:

Diagnostic-Guided Repair Instruction

Edit the invalid rewrite directly, using the original claim as the semantic anchor and the AtomEval diagnosis as repair guidance. Minimally fix the diagnosed failure while preserving the rewrite's wording and structure. Restore any changed or missing truth-critical constraint, including entities, relations, numbers, dates, negation, exclusivity terms, roles, locations, and modifiers. Do not correct toward the evidence, replace the attacked proposition with evidence-supported facts, or add caveats or truth-status labels. Return one declarative sentence as JSON: `{"repair_claim": "..."}`.

F Additional Results and Case Studies

F.1 Full Surface Results

Table 7 reports surface-proxy values for successful attacks on the BERT-based verifier. GPT-4.1 generally yields higher SBERT similarity than Mistral-7B, while Adv.PPL varies substantially by attack family.

Attack	Gen.	SBERT↑	Adv.PPL↓
AdvAdd	GPT-4.1	0.903	65.59
AdvAdd	Mistral-7B	0.723	47.35
Colloquial	GPT-4.1	0.880	128.97
Colloquial	Mistral-7B	0.734	132.16
DeSePtion	GPT-4.1	0.803	38.61
DeSePtion	Mistral-7B	0.686	35.29
FactMix	GPT-4.1	0.753	78.96
FactMix	Mistral-7B	0.657	64.22
Omission	GPT-4.1	0.840	186.46
Omission	Mistral-7B	0.571	150.23

Table 7: Surface-proxy statistics for successful adversarial rewrites on the BERT-based verifier. SBERT reports the mean Sentence-BERT similarity between the original claim and the successful rewrite. Adv.PPL reports the mean GPT-2 perplexity of successful adversarial rewrites.

F.2 Full Diagnostic Result

Table 8 shows that invalid successful attacks are dominated by ScopeLoss, especially for omission-style rewrites, while evidence drift is most prominent in FactMix and colloquial generations.

F.3 Case Study

We provide a claim-centered case study illustrating how different attack strategies and generators can produce qualitatively different rewrites for the same original instance.

Original claim: "Danger UXB is a desk."

Evidence: "Danger UXB is a 1979 British ITV television series set during the Second World War developed by John Hawkesworth and starring Anthony Andrews as Lieutenant Brian Ash, an officer in the Royal Engineers."

Representative Zero-shot Attack Prompt

System instruction

```
You are an expert in adversarial evaluation for automated fact verification.

You will be given an Original Claim, and fixed Evidence.
Your task is to rewrite the claim according to the specified attack strategy.

General rules:
• Output exactly ONE rewritten claim.
• Output a single sentence whenever possible.
• Do not output explanations, labels, or prefixes.
• Keep the claim fluent, natural, and fact-checkable.
• Avoid copying the original claim or an evidence sentence verbatim.
• The evidence is fixed and must not be rewritten.
```

Input

```
Original Claim: {original_claim}
Evidence: {evidence}
```

Strategy instruction: Add.Unver./AdvAdd-style unsupported addition

```
Rewrite the original claim by preserving its main assertion and adding exactly one plausible
auxiliary detail. The added detail should be related to the original claim, but it should not
be directly stated or explicitly entailed by the fixed evidence. It should sound like a normal
background detail that would require additional verification beyond the provided evidence.

Important constraints:
• Preserve the original claim's main assertion.
• Add exactly one minor auxiliary detail.
• Do not copy the added detail from the evidence.
• Do not recombine multiple evidence facts; that belongs to fact mixing.
• Do not correct the original claim.
• Keep the rewritten claim as one fluent declarative sentence.

Rewritten claim:
```

Figure 3: Representative zero-shot prompt used for adversarial claim rewriting. The shared system instruction is combined with a strategy-specific rewriting instruction and the fixed claim–evidence input.

Verifier	Generator	Attack Family	N	Invalid	Valid	EvDrift	ScopeLoss	EvEnt	UnverAdd
BERT-based	GPT-4.1	ADD.UNVER./ADVADD	37	0.0 (0)	100.0 (37)	–	–	0.0 (0)	97.3 (36)
BERT-based	GPT-4.1	COLLOQUIAL/LEXICAL	42	19.0 (8)	81.0 (34)	50.0 (4)	100.0 (8)	2.9 (1)	20.6 (7)
BERT-based	GPT-4.1	DESEPTION-STYLE	189	9.5 (18)	90.5 (171)	22.2 (4)	94.4 (17)	32.8 (56)	71.9 (123)
BERT-based	GPT-4.1	GEM-STYLE FACTMIX	103	57.3 (59)	42.7 (44)	91.5 (54)	89.8 (53)	86.4 (38)	79.5 (35)
BERT-based	GPT-4.1	OMISSION-STYLE	83	94.0 (78)	6.0 (5)	16.7 (13)	98.7 (77)	0.0 (0)	20.0 (1)
BERT-based	Mistral-7B	ADD.UNVER./ADVADD	129	67.4 (87)	32.6 (42)	48.3 (42)	95.4 (83)	30.9 (13)	100.0 (42)
BERT-based	Mistral-7B	COLLOQUIAL/LEXICAL	85	62.4 (53)	37.6 (32)	73.6 (39)	86.8 (46)	25.0 (8)	18.8 (6)
BERT-based	Mistral-7B	DESEPTION-STYLE	298	83.2 (248)	16.8 (50)	63.3 (157)	94.0 (233)	72.0 (36)	78.0 (39)
BERT-based	Mistral-7B	GEM-STYLE FACTMIX	194	75.8 (147)	24.2 (47)	81.6 (120)	85.0 (125)	85.1 (40)	83.0 (39)
BERT-based	Mistral-7B	OMISSION-STYLE	165	100.0 (165)	0.0 (0)	41.8 (69)	99.4 (164)	–	–
Gemma-7B	GPT-4.1	ADD.UNVER./ADVADD	11	0.0 (0)	100.0 (11)	–	–	0.0 (0)	100.0 (11)
Gemma-7B	GPT-4.1	COLLOQUIAL/LEXICAL	64	85.9 (55)	14.1 (9)	81.8 (45)	67.3 (37)	11.1 (1)	22.2 (2)
Gemma-7B	GPT-4.1	DESEPTION-STYLE	30	6.7 (2)	93.3 (28)	50.0 (1)	100.0 (2)	35.7 (10)	60.7 (17)
Gemma-7B	GPT-4.1	GEM-STYLE FACTMIX	62	50.0 (31)	50.0 (31)	100.0 (31)	83.9 (26)	96.8 (30)	67.7 (21)
Gemma-7B	GPT-4.1	OMISSION-STYLE	154	94.8 (146)	5.2 (8)	24.0 (35)	99.3 (145)	0.0 (0)	12.5 (1)
Gemma-7B	Mistral-7B	ADD.UNVER./ADVADD	157	84.1 (132)	15.9 (25)	55.3 (73)	89.4 (118)	48.0 (12)	96.0 (24)
Gemma-7B	Mistral-7B	COLLOQUIAL/LEXICAL	152	88.2 (134)	11.8 (18)	85.1 (114)	85.8 (115)	66.7 (12)	38.9 (7)
Gemma-7B	Mistral-7B	DESEPTION-STYLE	256	87.1 (223)	12.9 (33)	70.9 (158)	92.8 (207)	78.8 (26)	78.8 (26)
Gemma-7B	Mistral-7B	GEM-STYLE FACTMIX	139	88.5 (123)	11.5 (16)	91.1 (112)	86.2 (106)	87.5 (14)	62.5 (10)
Gemma-7B	Mistral-7B	OMISSION-STYLE	294	99.7 (293)	0.3 (1)	45.4 (133)	99.0 (290)	–	–

Table 8: AtomEval diagnostics for raw successful attacks. Percentages are shown with counts in parentheses. Invalid and Valid denote rewrites that fail or pass the binary gate; valid-side diagnostics are omitted for fewer than five valid rewrites.

Strategy	Gen.	Adversarial Rewrite	Attack	Observation	Explanation
LEXICAL	GPT-4.1	"Danger UXB is a piece of furniture."	Failure	ScopeLoss	The rewrite generalizes "desk" to "piece of furniture." It weakens the original proposition and does not flip the verifier in this instance.
	Mistral	"Danger UXB's actually a telly show, mate. It's a 1979 British series, see? Set during WW2, and starring Tony Andrews as an officer in the Royal Engineers."	Success	EvDrift	The rewrite replaces the original false desk proposition with the evidence-supported fact that *Danger UXB* is a television series.
DESEPTION	GPT-4.1	"Danger UXB, which shares its name with a well-known British television series, refers to a type of desk introduced in the late 20th century."	Success	Valid + EvEnt + UnverAdd	The rewrite preserves the original false desk proposition while adding evidence-supported context about the television series.
	Mistral	"During the Second World War, there existed a British television series titled Danger UXB, developed by John Hawkesworth and starring Anthony Andrews as Lieutenant Brian Ash, whose duties were not limited solely to a desk."	Success	EvDrift + ScopeLoss	The rewrite shifts toward the evidence-supported television-series description and weakens the desk proposition into a peripheral phrase.
FACTMIX	GPT-4.1	"Danger UXB is a desk used by the Royal Engineers during the Second World War."	Failure	Valid + UnverAdd	The rewrite preserves the false desk proposition but adds unsupported details blended from evidence-related entities and setting information. It does not flip the verifier here.
	Mistral	"Anthony Andrews, known for his role in the 1979 ITV series Danger UXB, appeared as a desk officer in the Royal Engineers during the Second World War."	Success	EvDrift + ScopeLoss	The rewrite shifts the subject from *Danger UXB* to Anthony Andrews, drops the original desk proposition, and mixes evidence-supported details with an unsupported desk-related role.
OMISSION	GPT-4.1	"Danger UXB is a television series."	Success	EvDrift	The rewrite directly replaces the original false desk proposition with the evidence-supported television-series fact.
	Mistral	"A device used for detecting and dealing with explosives is often a TV series setting."	Success	ScopeLoss	The rewrite generalizes away from the named entity and the original desk proposition, so the attacked proposition is no longer recoverable.
ADVADD	GPT-4.1	"Danger UXB is a desk that was once used in a British government office."	Failure	Valid + EvEnt + UnverAdd	The rewrite preserves the original false desk proposition while adding an unsupported office-use detail.
	Mistral	"Danger UXB was frequently used on the battlefields of Europe during the Second World War."	Failure	ScopeLoss	The rewrite drops the original desk proposition and introduces an unsupported battlefield-use interpretation of *Danger UXB*; it is semantically altered but does not flip the verifier.

Table 9: Claim-centered case study under fixed evidence. Strategy abbreviations correspond to the five attack families in Section 4.2: Lexical, DeSePtion, FactMix, Omission, and AdvAdd. The examples show that attack success and proposition preservation are distinct: some successful attacks are invalid because they drift toward evidence or lose the original proposition, while valid rewrites may still introduce evidence-supported or unsupported additional content. Diagnostic tags such as EvEnt and UnverAdd characterize these expansions but do not affect VASR.