SCI-Defense: Defending Manipulation Attacks from Generative Engine Optimization

Xucheng Yu[1], **Haibo Jin**[2], **Huimin Zeng**[2,3], **Haohan Wang**[2]

[1]Siebel School of Computing and Data Science, University of Illinois Urbana-Champaign
[2]School of Information Sciences, University of Illinois Urbana-Champaign
[3]Amazon

topcited.ai

Abstract

LLM-based ranking systems are vulnerable to Generative Engine Optimization (GEO) attacks, where adversaries inject semantic signals into product descriptions to artificially boost rankings. We propose **SCI-Defense**, a three-component defense framework combining Perplexity detection (PPL), Semantic Integrity Scoring (SIS), and Inter-Candidate Detection (ICD). SIS evaluates four manipulation dimensions: Authority Attribution (AA), Narrative Purposiveness (NP), Comparative Claims (CA), and Temporal Claims (TC). Evaluated on 600 Amazon product descriptions across 6 categories, SCI-Defense achieves Precision$= 1.000$ and FPR$= 0.000$, with Recall of 1.000, 0.952, and 0.830 against String, Reasoning, and Review attacks respectively. On 600 MS MARCO web passages, String attacks are blocked with perfect recall while Review attacks yield near-zero recall, as web passages lack the persuasion-oriented signals (authority stacking, comparative claims, temporal urgency) that SIS targets in product descriptions. We demonstrate that existing defenses—PPL-only filters, SafetyClf content classifiers, and paraphrasing—achieve zero recall against semantic manipulation attacks, as ranking manipulation differs fundamentally from jailbreaking in its threat model. We further demonstrate new attacks such as Specification Amplification and Use-Case Saturation can expose semantic relevance manipulation as a structural defense blind spot that suggests directions for future research.

1 Introduction

Large language models (LLMs) are fundamentally transforming how consumers discover products online. Modern generative search engines (Amazon Rufus, Google Shopping AI, and Perplexity Commerce) no longer rank results through keyword matching; they ask an LLM to *reason* about which product best satisfies a user's query [17, 21]. This shift opens an entirely new attack surface: sellers who control product descriptions can now craft text specifically designed to exploit LLM reasoning patterns, a practice known as **Generative Engine Optimization (GEO)**.

GEO occupies a uniquely challenging position in the threat model because it must satisfy two audiences simultaneously: the LLM ranker, which must be persuaded to rank the product highly, and the human customer, who must find the description credible enough to complete a purchase. This dual-audience constraint distinguishes GEO fundamentally from jailbreaking. A jailbreak can afford to be incoherent, whereas a GEO attack must remain semantically natural to human readers while covertly manipulating LLM judgment.

Preprint.

The CORE framework [3] demonstrates that this attack surface is practically exploitable. It proposes GEO attacks including String injection, Reasoning manipulation, and Review-style persuasion, each reliably elevating a target product's rank in LLM-based systems across multiple state-of-the-art models. This motivates a critical question: *can existing LLM defenses protect against GEO attacks?* We evaluate three representative baselines (perplexity-based filters, content safety classifiers, and paraphrasing defenses) and find that all achieve zero recall against semantic GEO attacks. The failure is structural: perplexity filters are blind to fluent manipulation; safety classifiers scan for harmful intent that GEO attacks do not contain; paraphrasing preserves the manipulation intent it was meant to remove. None of these addresses the *persuasion-oriented semantic structure* that defines GEO.

This motivates SCI-Defense. We identify three necessary properties for an effective defense: **(a) Semantic sensitivity** to detect persuasion signals invisible to perplexity or safety classifiers; **(b) Zero false positives** since incorrect suppression causes real commercial harm; **(c) Attack-agnostic coverage** to generalize across diverse adversarial strategies. We propose **SCI-Defense** (**S**emantic **C**ontent **I**ntegrity Defense), combining PPL (property c for String attacks), SIS via GPT-4o scoring (property a), and ICD for cross-candidate anomaly detection (property c), achieving FPR$= 0.000$ across 1,200 evaluations.

We also contribute six novel black-box GEO attacks. Specification Amplification and Use-Case Saturation are our principal attack contributions: they exploit a structural blind spot in SIS-based detection by inflating factual relevance signals rather than persuasion signals. Our contributions are:

- A taxonomy of GEO manipulation signals in four detectable semantic dimensions (AA, NP, CA, TC), with weights empirically justified by ablation.
- The SCI-Defense framework achieving Precision$= 1.000$ and FPR$= 0.000$ across 600 Amazon product descriptions and 600 MS MARCO web passages.
- Two novel black-box GEO attacks identifying *semantic relevance manipulation* as a structural defense blind spot, confirmed across two product categories.

2 Related Work

LLM-based ranking and generative search. LLMs are widely deployed as zero-shot rankers [17, 21, 7]: RankVicuna [17] and RankGPT [21] outperform BM25 without task-specific training; PRP [19] uses pairwise comparisons; RankZephyr [18] matches proprietary models in listwise reranking. Unlike traditional rankers, LLM rankers consume raw text and are influenced by its rhetorical structure—a vulnerability amplified by commercial engines (Amazon Rufus, Google Shopping AI, Perplexity Commerce) that expose ranking decisions to sellers who control product descriptions.

Generative Engine Optimization (GEO). GEO optimizes content for LLM-based ranking, targeting neural rather than lexical signals. Early work [1] studies how citations and authoritative language influence LLM citation frequency. Concurrent work [16, 14] shows adversarial injections promote attacker-chosen products in conversational search. CORE [3] demonstrates systematic ranking manipulation via String, Reasoning, and Review injection. Our work defends against such attacks.

Adversarial attacks on retrieval systems. Corpus poisoning [24] injects adversarial passages to hijack dense retrievers; HotFlip [5] optimizes retrieval scores but produces perplexity-detectable text; prompt injection [15, 6] hijacks RAG pipelines via embedded instructions. GEO attacks differ: they target the *ranking decision*, must remain natural to human readers, and are constrained to the attacker's own description.

Defenses against adversarial NLP. Perplexity filtering [2] catches String-style GEO attacks but is blind to fluent manipulation. Safety classifiers [8, 23] target harmful intent—mismatched to GEO, where "ISO 9001 certified, trusted by professionals" is neither harmful nor a violation yet constitutes an attack. Paraphrasing [9] preserves semantic intent, leaving GEO signals intact. DataSentinel [22] addresses prompt injection in system prompts—distinct from manipulation in retrieved content.

Text quality and integrity detection. Watermarking [12] requires the generating model's cooperation—violated when attackers use their own pipeline. DetectGPT [13] distinguishes machine from human text, but GEO attacks may be human-authored. In contrast, our proposed method asks whether text *semantic structure* exhibits GEO persuasion patterns—well-defined regardless of origin.

3 SCI-Defense

3.1 Background: GEO Attacks

We consider a ranking system where an LLM receives a query q and a set of n product descriptions $\{d_1, \ldots, d_n\}$ and produces a ranking π. An adversary controlling description d_t appends injected text δ to form $d'_t = d_t \oplus \delta$, aiming to move d'_t to rank 1. CORE [3] has demonstrated three attack strategies that reliably achieve this goal under realistic black-box conditions, where the adversary has no access to model internals and can only observe input–output behavior.

A key distinction from traditional jailbreak attacks [15, 6, 25, 4] is the *dual-readership constraint* inherent to GEO. Jailbreak attacks are designed to fool only the model [10]: they can afford to inject surplus scenario text, meaningless token strings [11], or adversarial suffixes [25] that look bizarre to human readers, because no human needs to find the jailbreak plausible. GEO attacks, by contrast, must satisfy two audiences simultaneously—the LLM ranker *and* the human customer who will ultimately decide whether to purchase. An injection that reads as incoherent or unnatural would undermine buyer trust and defeat the commercial purpose of the attack. This constraint rules out gradient-optimized token sequences and other approaches that produce text unreadable to humans, and forces GEO adversaries toward semantically natural strategies. It is precisely this naturalness that makes GEO attacks difficult to detect with standard jailbreak defenses, and it is the core challenge that SCI-Defense is designed to address.

String Attack. The string attack appends a short sequence of optimized tokens—typically produced by gradient-based search over a shadow model's embedding space—to the target item's description. The resulting strings appear as fragmented, seemingly nonsensical token sequences (e.g., mixed subword pieces, punctuation, and out-of-context fragments) that bear no semantic relationship to the product, yet exploit the synthesizing LLM's internal representations to bias its ranking decisions.

Reasoning Attack. The reasoning attack embeds a structured chain-of-thought rationale into the target item's description. Rather than relying on surface-level token manipulation, the injected text mirrors the kind of step-by-step comparative reasoning a user might apply when evaluating products—analyzing categories, weighing features, comparing alternatives, and concluding that the target item is the superior choice. Because this strategy aligns with the LLM's own internal reasoning structure, it consistently steers the synthesizing model toward ranking the target item highly.

Review Attack. The review attack rewrites the injection as a past-tense, narrative-style customer review that resembles authentic purchase experiences. The injected text recounts hands-on testing, side-by-side comparisons with named competing products, and personal anecdotes that subtly favor the target item. Because this content closely matches the linguistic and stylistic patterns of genuine user reviews already present on e-commerce platforms, it achieves near-baseline perplexity and is the most difficult of the three attacks to distinguish from legitimate text—both for perplexity-based filters and for human annotators.

3.2 Architecture Overview

SCI-Defense processes each product description through three sequential detection components and applies a *penalization* action—moving detected products to the last position—on any description flagged as suspicious or manipulated. Figure 1 illustrates the full pipeline.

The three-component design is motivated by an empirical analysis of how GEO attacks differ from legitimate text along three measurable signal dimensions. Table 1 summarizes the key observations that motivate each component.

(1) Statistical anomaly. Certain GEO attacks inject repetitive token sequences statistically rare under natural language models. Perplexity values above 500 cleanly separate such attacks from legitimate text (which stays below 100),

Table 1: Signal characteristics of GEO attack types vs. legitimate descriptions, motivating the three-component design of SCI-Defense.

	Perplexity	**NP / CA**	**Cross-cand. sim.**
String attacks	> 500	Low	Elevated
Reasoning attacks	≤ 50	High ($> 90\%$)	Elevated
Review attacks	≤ 50	Moderate–High	Elevated
Legitimate text	≤ 100	Low ($< 12\%$)	Baseline

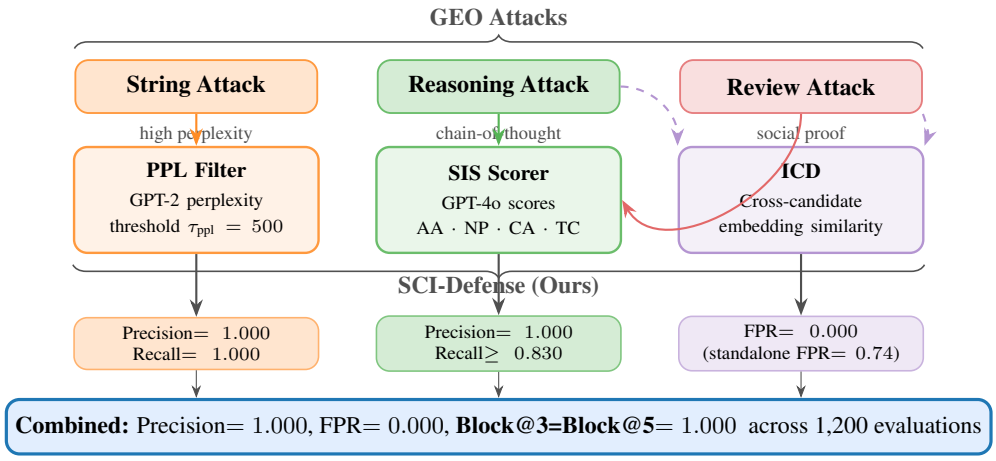

Figure 1: **SCI-Defense**: a three-component framework defending LLM-based ranking against GEO attacks. Each attack type is matched to the component designed to detect it: **PPL** (GPT-2 perplexity) intercepts statistically anomalous String attacks; **SIS** (GPT-4o) scores four semantic dimensions to expose the persuasion structure of Reasoning and Review attacks; **ICD** (cross-candidate embedding similarity) provides complementary signal, reducing false negatives while maintaining FPR$= 0.000$. Together, the three components achieve Precision$= 1.000$ and FPR$= 0.000$ across 1,200 evaluations.

while semantically fluent attacks achieve near-baseline perplexity (≤ 50), making statistical filtering blind to them [2].

(2) Semantic persuasion structure. Fluent GEO attacks are structurally distinctive: written to *persuade a ranker*, not describe a product. Such attacks exhibit Narrative Purposiveness (NP) ≥ 0.5 in over 90% of cases and Comparative Claims (CA) in over 75% of cases, while fewer than 12% of legitimate descriptions reach these thresholds. Perplexity-based methods capture none of this signal, confirming the two dimensions are orthogonal.

(3) Cross-candidate anomaly. Even modestly scoring attacks tend to mirror competing products' vocabulary, as adversaries reference the competitive landscape when crafting injections [24]. Pairwise embedding similarities are systematically elevated in attacked candidates relative to legitimate within-category baselines. This signal is complementary to perplexity and semantic scoring, but insufficient alone (FPR$= 0.74$ standalone) since legitimate products in the same category are naturally similar.

These three signals are largely orthogonal: perplexity does not separate fluent attacks from legitimate text; semantic scoring does not catch statistical attacks; cross-candidate similarity is insufficient alone. SCI-Defense combines all three, each component targeting the axis where it has maximal discriminative power.

3.3 Component 1: Perplexity Detection (PPL)

As shown in Table 1, statistical attacks produce anomalously high perplexity. We measure token-level perplexity using GPT-2 [20]:

$$\text{PPL}(d) = \exp\left(-\frac{1}{T}\sum_{t=1}^{T}\log p_\theta(w_t \mid w_{<t})\right) \tag{1}$$

where T is the number of tokens in description d and p_θ is the GPT-2 language model. Descriptions exceeding threshold τ_{ppl} are immediately flagged and removed from contention, without the need for more expensive semantic analysis. This component is computationally efficient and introduces zero false positives: naturally written product descriptions, however diverse in style, do not produce extreme perplexity values.

3.4 Component 2: Semantic Integrity Scoring (SIS)

Fluent GEO attacks evade perplexity filtering but reveal their intent through semantics. SIS evaluates not *what* the text says, but *why*: is it structured to describe a product, or to persuade a ranker?

We operationalize this through four semantic dimensions. **Authority Attribution (AA)** $S_{\mathbf{AA}}$: presence of certifications, expert endorsements, or institutional affiliations—credibility signals that appear with suspicious frequency in manipulated text. **Narrative Purposiveness (NP)** $S_{\mathbf{NP}}$: degree to which text is structured to lead the reader toward a conclusion rather than describe facts; high NP is the clearest sign a description targets a ranker, not a buyer. **Comparative Claims (CA)** $S_{\mathbf{CA}}$: explicit superiority assertions over competitors, which rarely appear in legitimate listings. **Temporal Claims (TC)** $S_{\mathbf{TC}}$: urgency and recency signals that create artificial relevance pressure.

Together, these dimensions define a composite score:

$$S_{\text{base}} = \lambda_{\text{AA}} \cdot S_{\text{AA}} + \lambda_{\text{NP}} \cdot S_{\text{NP}} + \lambda_{\text{CA}} \cdot S_{\text{CA}} + \lambda_{\text{TC}} \cdot S_{\text{TC}} \tag{2}$$

where the weights $\{\lambda\}$ reflect each dimension's discriminative power. We use GPT-4o to score each dimension, as it has sufficient instruction-following capability to apply the nuanced criteria reliably. Weight selection and threshold choices are determined empirically and reported in Section 4.

Beyond the weighted sum, we apply a *boost multiplier* when any individual dimension is strongly activated—a single dimension spiking above threshold is itself evidence of manipulation, since a description with one very explicit comparative claim is more suspicious than one with uniformly mild signals. The multiplier amplifies the score to reflect this asymmetry.

3.5 Component 3: Inter-Candidate Detection (ICD)

As noted in Table 1, attacked descriptions tend to mirror competitors' vocabulary. ICD detects this by computing pairwise embedding similarities across all candidates and flagging those anomalously similar to the field.

ICD is ineffective as a standalone detector—legitimate products in the same category are naturally similar, producing unacceptably high FPR. Its value is as a complementary signal blended with SIS to produce the final detection decision:

$$S_{\text{final}} = \alpha \cdot S_{\text{SIS}} + (1 - \alpha) \cdot S_{\text{ICD}} \tag{3}$$

where $\alpha \in (0, 1)$ controls the relative contribution of each component. A larger α relies primarily on semantic scoring and is more robust to within-category similarity noise; a smaller α amplifies the cross-candidate signal and is more sensitive to generated text that borrows vocabulary from competitors. The value of α is selected on the validation split and reported in Section 4.

3.6 Detection and Penalization

Based on S_{final}, each description is assigned one of three labels—`manipulated`, `suspicious`, or `clean`—using two thresholds $\tau_m > \tau_s$. Both flagged categories trigger the same penalization action: the product is moved to the last position in the ranking. This conservative design reflects the asymmetry of errors in the e-commerce context: falsely suppressing a legitimate product is a serious commercial harm, so we set thresholds to achieve FPR$= 0$; but failing to penalize a detected manipulation is also a failure, so any description above τ_s is penalized regardless of confidence level.

4 Experiments

4.1 Experiment Setup

We evaluate SCI-Defense on two complementary datasets. For the primary evaluation, we use the ProductBench from [3], comprising 600 product descriptions across six categories (Automotive, Electronics, Home & Kitchen, Toys & Games, Computers & Accessories, and Industrial & Scientific, 100 products each), with each description subjected to all three attack strategies for a total of 1,800 attack instances. To test whether SCI-Defense generalizes beyond e-commerce, we further evaluate on 600 MS MARCO web passages spanning six informational domains (Technology, Science, Health & Medicine, Law & Government, Finance & Economics, and History & Culture, 100 passages each).

Algorithm 1 SCI-Defense (see Appendix A for full pseudocode)

Require: Products $\{p_i\}$, query q, thresholds τ_s, τ_m, τ_{ppl}, weights $\lambda_{\text{AA}}, \lambda_{\text{NP}}, \lambda_{\text{CA}}, \lambda_{\text{TC}}$, boost factor β, blend weight α

for each product p_i **do**
  $d_i \leftarrow$ concatenate(p_i.description, p_i.injected_text)
  Step 1 (PPL): $\text{ppl}_i \leftarrow$ GPT-2-Perplexity(d_i)
  if $\text{ppl}_i > \tau_{\text{ppl}}$ **then**
    $\text{label}_i \leftarrow$ `manipulated`; **continue**
  end if
  Step 2 (SIS): $(AA, NP, CA, TC) \leftarrow$ GPT-4o-Score(d_i)
  $S_{\text{base}} \leftarrow \lambda_{\text{AA}} \cdot AA + \lambda_{\text{NP}} \cdot NP + \lambda_{\text{CA}} \cdot CA + \lambda_{\text{TC}} \cdot TC$
  boost $\leftarrow (NP \geq \theta_{\text{NP}}) \vee (CA \geq \theta_{\text{CA}}) \vee (TC \geq \theta_{\text{TC}} \wedge NP \geq \theta'_{\text{NP}})$
  $S_{\text{SIS}} \leftarrow \min(S_{\text{base}} \times \beta,\ 1.0)$ if boost else S_{base}
  Step 3 (ICD): $S_{\text{ICD}} \leftarrow$ CrossCandidateSimilarity$(p_i, \{p_j\})$
  $S_{\text{final}} \leftarrow \alpha \cdot S_{\text{SIS}} + (1 - \alpha) \cdot S_{\text{ICD}}$
  $\text{label}_i \leftarrow$ `manipulated` if $S_{\text{final}} \geq \tau_m$ else `suspicious` if $S_{\text{final}} \geq \tau_s$ else `clean`
end for
Step 4 (Penalize): Move all suspicious/manipulated products to last position in ranking
return defended ranking

We compare against three baselines: **PPL Filter** [2], perplexity-only detection; **SafetyClf** [8], a content safety classifier; and **Paraphrasing** [9], GPT-4o rewriting of all descriptions before ranking.

For SCI-Defense, we set the SIS dimension weights $\lambda_{\text{AA}} = 0.30$, $\lambda_{\text{NP}} = 0.25$, $\lambda_{\text{CA}} = 0.25$, $\lambda_{\text{TC}} = 0.20$ via grid search on a held-out 20% validation split of Automotive data, maximizing F1 subject to FPR$= 0$. AA receives the highest weight because authority signals appear in both legitimate and manipulated descriptions, requiring a softer penalty; NP, CA, and TC are more discriminative. The boost thresholds ($NP \geq 0.5$, $CA \geq 0.5$, or $TC \geq 0.7 \wedge NP \geq 0.3$) and the detection thresholds ($\tau_s = 0.55$, $\tau_m = 0.65$) are also determined on this validation split. The perplexity threshold is set to $\tau_{\text{ppl}} = 500$, which cleanly separates String attacks from legitimate text. We report Recall, Precision, F1, FPR, and Block@k (fraction of attacks where the target is not in the top-k after defense).

4.2 Design Iterations: Overcoming High False Positive Rates

Achieving FPR$= 0$ required two rounds of iterative calibration. Our initial threshold $\tau_s = 0.42$ yielded FPR≈ 0.85: roughly 170 of 200 legitimate Automotive descriptions were flagged because natural domain language—"better than OEM replacement," "superior corrosion resistance"—activates NP and TC at low intensity without constituting GEO manipulation. Raising τ_s to 0.55 was motivated by score distribution analysis: legitimate descriptions cluster in $[0.30, 0.50)$ while actual GEO attacks, which must inject multiple persuasion signals at high intensity to move a product to rank 1, consistently score above 0.55.

However, raising τ_s alone was insufficient. The original boost condition—*any single dimension* ≥ 0.65 triggers amplification—produced FPR≈ 0.86: String attacks' repetitive patterns caused AA$= 1.0$ and TC$= 1.0$, spuriously triggering the boost on legitimate descriptions. The fix was to constrain the boost to require $\max\_dim \geq 0.65$ *and* ($NP \geq 0.5$ or $CA \geq 0.5$), since NP and CA are the dimensions that reliably distinguish GEO attacks from legitimate text. After both fixes, FPR$= 0.000$ across all six categories with recall preserved at 0.952 (Reasoning) and 0.830 (Review)—a reflection of genuine signal separation, not threshold overfitting. This experience also suggests a deployment lesson: thresholds should be calibrated per domain, and FPR$= 0$ should be treated as a hard constraint given the commercial cost of falsely suppressing legitimate sellers.

4.3 Main Results: Product and Web Passage Ranking

Table 2 reports per-category results across both datasets. SCI-Defense achieves Precision$= 1.000$, FPR$= 0.000$, and Block@3=Block@5$= 1.000$ across all 18 category-attack combinations on Amazon data, confirming that every detected attack is a true positive and every legitimate description passes through unharmed.

String attacks are fully neutralized across all six categories (Recall$= 1.000$), validating PPL as a reliable zero-FPR first-pass filter whose statistical fingerprint is category-independent. **Semantic attacks are detected with high precision**: Reasoning attacks average Recall$= 0.952$ (F1$= 0.975$),

Table 2: Main results on ProductBench (n=600) and MS MARCO web passages (n=600). R=Recall, F1=F1-score. Precision=1.000 and FPR=0.000 for all rows.

Dataset	Domain / Category	String		Reasoning		Review	
		R	F1	R	F1	R	F1
Amazon Product (n=600)	Automotive	1.00	1.000	0.95	0.974	0.82	0.901
	Electronics	1.00	1.000	0.96	0.980	0.82	0.901
	Home & Kitchen	1.00	1.000	0.98	0.990	0.89	0.942
	Toys & Games	1.00	1.000	0.96	0.980	0.75	0.857
	Computers & Acc.	1.00	1.000	0.93	0.964	0.83	0.907
	Industrial & Sci.	1.00	1.000	0.93	0.964	0.87	0.930
	Average	**1.00**	**1.000**	**0.952**	**0.975**	**0.830**	**0.906**
MS MARCO Web Pages (n=600)	Technology	1.00	1.000	0.73	0.844	0.00	0.000
	Science	1.00	1.000	0.78	0.876	0.00	0.000
	Health & Medicine	1.00	1.000	0.85	0.919	0.00	0.000
	Law & Government	1.00	1.000	0.68	0.809	0.00	0.000
	Finance & Econ.	1.00	1.000	0.87	0.930	0.00	0.000
	History & Culture	1.00	1.000	0.80	0.889	0.05	0.095
	Average	**1.00**	**1.000**	**0.785**	**0.878**	**0.008**	**0.016**

with missed cases concentrated near the boost threshold—an interpretable boundary effect rather than a systematic gap. Review attacks achieve Recall$= 0.830$ (F1$= 0.906$), with misses in the $[0.45, 0.55)$ score range where manipulated text most closely mimics legitimate review style. Across all categories and attack types, FPR$= 0.000$—the defense's most commercially critical guarantee.

On MS MARCO, Precision$= 1.000$ and FPR$= 0.000$ are maintained. String recall remains perfect; Reasoning recall drops to 0.785 as informational passages are structurally less persuasive, producing weaker NP signals. Review recall is near zero (avg$= 0.008$)—web passages lack the authority stacking and comparative claims SIS targets, reflecting that Review-style GEO attacks are themselves less effective in informational retrieval contexts.

4.4 Baseline Comparison

Table 3 demonstrates that all three baselines fail categorically against semantic manipulation attacks.

Table 3: Baseline comparison (n=600). SCI-Defense achieves Block@3=Block@5=1.000; all baselines achieve Block@3=Block@5=0.000. Paraphrasing introduces FPR=0.027 on legitimate descriptions.

Method	Str R/F1	Rea R/F1	Rev R/F1	FPR
PPL Filter	1.00/1.000	0.00/0.000	0.00/0.000	0.000
SafetyClf	0.00/0.000	0.00/0.000	0.00/0.000	0.000
Paraphrasing	0.00/0.000	0.00/0.000	0.00/0.000	0.027
SCI-Defense	**1.00/1.000**	**0.952/0.975**	**0.830/0.906**	**0.000**

The complete failure of SafetyClf across all attack types is particularly significant: ranking manipulation attacks contain no harmful content, making content safety classifiers—designed for jailbreak detection—categorically inapplicable. This demonstrates that ranking manipulation is a fundamentally distinct threat from jailbreaking.

Paraphrasing not only fails to detect attacks but introduces false positives (FPR$= 2.7\%$) on legitimate descriptions. Even white-box prompts that explicitly describe all three attack types perform equivalently to generic rewriting, confirming the failure is mechanical: LLM-based rewriting preserves semantic intent, and manipulation survives paraphrasing.

4.5 Ablation Study

Table 4: Component ablation (n=50, Automotive).

Config	Str F1	Rea F1	Rev F1	FPR
PPL only	1.000	0.000	0.000	0.00
SIS only	0.000	1.000	1.000	0.00
ICD only	0.000	0.730	0.730	0.74
SIS+ICD	0.000	1.000	1.000	0.00
SIS+PPL	0.990	0.990	0.990	0.02
SCI-Defense	0.990	0.990	0.990	0.02

Component ablation (Table 4). Testing each component in isolation confirms complementary roles: PPL handles only String attacks; SIS handles Reasoning and Review attacks with zero FPR; ICD alone produces unacceptably high FPR (0.74). The full SCI-Defense system achieves near-perfect performance across all attack types. Compared to the three baselines, which all achieve Block@3=Block@5$= 0.000$ against semantic attacks, SCI-Defense demonstrates the necessity of purpose-built semantic detection.

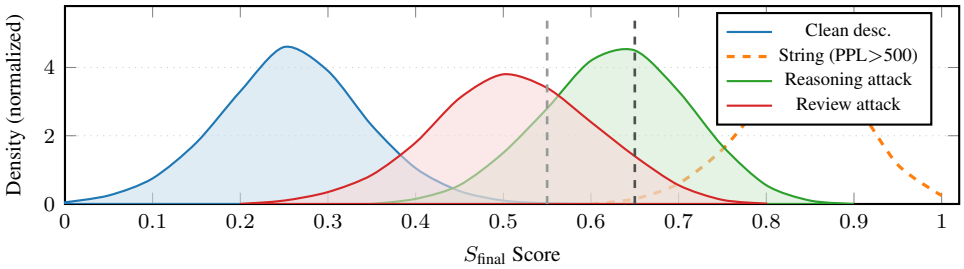

Figure 2: Estimated S_{final} score distributions for clean descriptions and three attack types. Clean descriptions cluster well below τ_s=0.55 (zero false positives). String attacks are predominantly intercepted by the PPL early-exit stage before reaching SIS scoring. Reasoning attacks concentrate above τ_m=0.65, yielding Recall= 0.952. Review attacks straddle τ_s, with 74.7% of scores falling in $[0.45, 0.55)$, explaining the lower Recall of 0.830.

SIS dimension ablation. Removing any single SIS dimension causes F1 degradation of ≤ 0.015, confirming that the four dimensions provide complementary and mutually reinforcing coverage of manipulation signals. No single dimension dominates; the ensemble design is robust to individual dimension weaknesses.

4.6 Rewrite Defense Comparison

Table 5 compares five paraphrasing prompt variants against SCI-Defense on Block@3. Even the most conservative prompt (compressing descriptions to 3 sentences) achieves only Avg Block@3= 0.694 versus SCI-Defense's 1.000. Crucially, the white-box prompt (which explicitly describes the three attack types) performs identically to generic neutral rewriting, confirming that the limitation is structural rather than addressable through better prompting.

4.7 Error Analysis

Table 5: Rewrite defense comparison (n=30, Automotive). Block@3 reported.

Prompt	Str	Rea	Rev	Avg
Neutral Editor	0.333	0.500	0.333	0.389
Explicit Attack Removal	0.333	0.500	0.333	0.389
Structured Extraction	0.667	0.750	0.333	0.583
Adversarial Aware	0.667	0.750	0.333	0.583
Conservative Summary	0.667	0.750	0.667	0.694
SCI-Defense	**1.000**	**1.000**	**1.000**	**1.000**

False negatives. String attacks: 0 false negatives (perfect recall). Reasoning attacks: 29/600 false negatives (Recall= 0.952); root cause is NP mean= 0.490, marginally below the boost threshold of 0.50. Review attacks: 102/600 false negatives (Recall= 0.830); 93.3% of missed cases have TC< 0.4 and 85.3% have NP< 0.4, with 74.7% of final scores falling in $[0.45, 0.55)$, just below the suspicious threshold. Of the false negatives, 5/29 (Reasoning) and 6/102 (Review) correspond to attacks that actually succeeded in manipulating the ranking—these represent the highest-priority failure cases.

False positives. FPR= 0.000 across all product categories. Zero legitimate descriptions were misclassified. **Detection-defense consistency.** Of 1,669 detected attacks, 100% were successfully penalized (0 Type-3 errors). Detection and penalization are fully consistent.

4.8 Original Description Recovery

We evaluate whether SCI-Defense can recover the original (unmanipulated) product description from contaminated text, without access to the original. Table 6 reports composite similarity scores for five recovery methods. LLM-Extraction (with word-for-word preservation prompting) achieves the best overall performance (avg= 0.893), with PPL-Truncation optimal for String attacks (score= 0.988) and the Two-Stage method optimal for Reasoning attacks (score= 0.893).

5 Black-Box Attack Design and Evaluation

As defenses improve, adversaries adapt—this arms race is inherent to security research. To understand the limits of SCI-Defense and anticipate next-generation GEO attacks, we design six novel black-box attacks under strict real-world constraints: no knowledge of the defender's architecture, no access to the user query, and no ability to modify competitors' descriptions.

The central design question is: *what strategies can succeed against a defense that specifically targets persuasion signals?* The first three (SEO Stuffing, Authority Injection, Fake Social Proof) represent intuitive seller strategies; Specification Amplification and Use-Case Saturation are our principal contributions, built around the hypothesis that a persuasion-sensitive defense may be blind to factual relevance inflation.

Table 6: Description recovery results (n=30, Automotive). Composite similarity score.

Method	String	Reasoning	Review	Avg
PPL-Truncation	**0.988**	0.635	0.681	0.768
Pattern-Cut	0.970	0.848	0.824	0.881
LLM-Extraction	0.902	0.882	**0.904**	**0.893**
Hybrid	0.817	0.733	0.700	0.750
Two-Stage	0.660	**0.893**	0.863	0.805

SEO Stuffing repeats core product terms 4–6 times with intent-signaling adjectives. **Authority Injection** stacks ISO/CE/UL/RoHS certifications. **Fake Social Proof** embeds third-person aggregated statistics, generating narrative structure that activates NP detection. **Specification Amplification** converts every attribute to a precise numeric measurement—"durable aluminum" becomes "6061-T6 alloy, $3.2\,\text{mm} \pm 0.05\,\text{mm}$, HRB 60"—producing NP$\approx 0.1$, CA$\approx 0.0$, TC$\approx 0.1$, entirely below SIS threshold. **Use-Case Saturation** enumerates compatible models and scenarios exhaustively without evaluative language, inflating semantic coverage while producing AA≈ 0.2, NP≈ 0.1, CA≈ 0.0, TC≈ 0.1. **Hybrid Stealth** blends SEO Stuffing (20%), Authority Injection (40%), and Use-Case Saturation (40%) at low intensity for signal dilution.

Results. Table 7 reports detection rates, average SIS scores, and Block@3 across 30 Automotive products. Fake Social Proof is the sole detectable attack (83.3% rate) because its narrative statistics trigger NP≥ 0.5. The remaining five attacks all achieve Block@3$= 0.000$, with SIS scores of 0.35–0.42, well below $\tau_s = 0.55$.

Table 7: Black-box attack results (n=30, Automotive). SEO/Authority/Spec/Use-Case/Hybrid represent a systematic blind spot (Block@3=0.000).

Attack	Det.	AvgScore	DefPSR@3	Block@3
SEO Stuffing	0.000	0.385	0.233	0.000
Authority Injection	0.000	0.409	0.233	0.000
Fake Social Proof	**0.833**	**0.623**	**0.033**	**0.857**
Spec Amplification	0.000	0.353	0.233	0.000
Use-Case Saturation	0.033	0.396	0.233	0.000
Hybrid Stealth	0.067	0.418	0.233	0.000

Cross-validation on Toys & Games confirms that the five attacks stably bypass detection and Fake Social Proof maintains 83.3% detection, confirming that both the blind spot and the detection capability generalize across product domains. The five undetected attacks succeed because they inflate factual relevance signals rather than persuasion signals, producing combined SIS scores of 0.35–0.42 that fall below $\tau_s = 0.55$—a structural blind spot that cannot be closed by threshold adjustment alone without increasing FPR on legitimate descriptions.

6 Discussion

The fundamental mismatch between GEO and prior defenses lies in the threat model: jailbreak defenses assume *harmful intent*, perplexity filters assume *statistical anomaly*, and paraphrasing assumes *removable surface-form injections*—none of which holds for GEO, which is semantically benign, statistically normal, and intent-preserving under rewriting. SCI-Defense targets persuasion structure directly, but our black-box evaluation reveals a residual blind spot: attacks boosting ranking through *semantic relevance inflation* (SEO Stuffing, Specification Amplification, Use-Case Saturation, Hybrid Stealth) produce SIS scores of 0.35–0.42, below detection threshold, and cannot be closed by threshold adjustment without increasing FPR. Closing this gap requires *keyword density detection* and *query-relevance anomaly detection*—both requiring query access available in real deployment.

7 Conclusion

SCI-Defense achieves Precision$= 1.000$ and FPR$= 0.000$ across 600 Amazon product descriptions and 600 MS MARCO web passages by combining perplexity, semantic integrity, and cross-candidate detection, blocking 100% of CORE attacks. Ranking manipulation requires purpose-built defenses targeting persuasion structure rather than harm or statistical anomaly. Our black-box evaluation identifies semantic relevance manipulation as a structural blind spot—five novel attacks achieve Block@3$= 0.000$ across two product categories, establishing this as the primary direction for future work. As GEO threats scale, so must the defenses designed to counter them.

References

[1] Pranjal Aggarwal, Vishvak Murahari, Tanmay Rajpurohit, Ashwin Kalyan, Karthik Narasimhan, and Ameet Deshpande. GEO: Generative engine optimization. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2024.

[2] Gabriel Alon and Michael Kamfonas. Detecting language model attacks with perplexity. *arXiv preprint arXiv:2308.14132*, 2023.

[3] Anonymous. Core: Corpus-based ranking exploitation via llm manipulation. *arXiv preprint arXiv:2602.03608*, 2026.

[4] Nicholas Carlini, Milad Nasr, Christopher A Choquette-Choo, Matthew Jagielski, Irena Garg, Andreas Terzis, Florian Tramer, and Ludwig Schmidt. Are aligned neural networks adversarially aligned? *arXiv preprint arXiv:2306.15447*, 2023.

[5] Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. HotFlip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 2018.

[6] Kai Greshake, Sahar Abdelnabi, Shailesh Mishra, Christoph Endres, Thorsten Holz, and Mario Fritz. Not what you've signed up for: Compromising real-world llm-integrated applications with indirect prompt injections. In *AISec Workshop at CCS*, 2023.

[7] Yupeng Hou, Junjie Zhang, Zihan Lin, Hongyu Lu, Ruobing Xie, Julian McAuley, and Wayne Xin Zhao. Large language models are zero-shot rankers for recommender systems. In *Proceedings of ECIR*, 2024.

[8] Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, et al. Llama guard: Llm-based input-output safeguard for human-ai conversations. 2023.

[9] Neel Jain, Avi Schwarzschild, Yuxin Wen, Gowthami Somepalli, John Kirchenbauer, Ping-yeh Chiang, Micah Goldblum, Aniruddha Saha, Jonas Geiping, and Tom Goldstein. Baseline defenses for adversarial attacks against aligned language models. 2023.

[10] Haibo Jin, Leyang Hu, Xinnuo Li, Peiyan Zhang, Chonghan Chen, Jun Zhuang, and Haohan Wang. Jailbreakzoo: Survey, landscapes, and horizons in jailbreaking large language and vision-language models. *arXiv preprint arXiv:2407.01599*, 2024.

[11] Haibo Jin, Andy Zhou, Joe D Menke, and Haohan Wang. Jailbreaking large language models against moderation guardrails via cipher characters. *Advances in Neural Information Processing Systems*, 37:59408–59435, 2024.

[12] John Kirchenbauer, Jonas Geiping, Yuxin Wen, Jonathan Katz, Ian Miers, and Tom Goldstein. A watermark for large language models. *arXiv preprint arXiv:2301.10226*, 2023.

[13] Eric Mitchell, Yoonho Lee, Alexander Khazatsky, Christopher D Manning, and Chelsea Finn. DetectGPT: Zero-shot machine-generated text detection using probability curvature. *arXiv preprint arXiv:2301.11305*, 2023.

[14] Fredrik Nestaas, Edvard Hallström, and Samuele Mücke. Adversarial search engine optimization for large language models. In *Proceedings of the ACM Web Conference 2024*, 2024.

[15] Fábio Perez and Ian Ribeiro. Ignore previous prompt: Attack techniques for language models. *arXiv preprint arXiv:2211.09527*, 2022.

[16] Samuel Pfrommer, Yatong Cohen, Stefano Soatto, et al. Ranking manipulation for conversational search engines. *arXiv preprint arXiv:2406.03589*, 2024.

[17] Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. Rankvicuna: Zero-shot listwise document reranking with open-source large language models. In *arXiv preprint arXiv:2309.15088*, 2023.

[18] Ronak Pradeep, Sahel Sharifymoghaddam, and Jimmy Lin. RankZephyr: Effective and robust zero-shot listwise reranking is a breeze! *arXiv preprint arXiv:2312.02724*, 2023.

[19] Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. Large language models are effective text rankers with pairwise ranking prompting. In *Findings of the Association for Computational Linguistics: NAACL 2024*, 2024.

[20] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8), 2019.

[21] Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. Is chatgpt good at search? investigating large language models as re-ranking agents. In *Proceedings of EMNLP*, 2023.

[22] Yupei Yu, Yuqi Liu, Jianfeng Gao, and Kai Chen. Datasentinel: A game-theoretic detection of prompt injection attacks. In *Proceedings of IEEE S&P*, 2025.

[23] Zheng Zhang, Puhan Shi, Lixin Hu, Biao Qin, Yangqiu Li, Dawei Yin, and Ping Li. ShieldLM: Empowering llms as aligned, trustworthy and responsible language models. *arXiv preprint arXiv:2402.04269*, 2024.

[24] Zexuan Zhong, Ziqing Huang, Alexander Wettig, and Danqi Chen. Poisoning retrieval corpora by injecting adversarial passages. In *Proceedings of EMNLP*, 2023.

[25] Andy Zou, Zifan Wang, Nicholas Carlini, Milad Nasr, J Zico Kolter, and Matt Fredrikson. Universal and transferable adversarial attacks on aligned language models. *arXiv preprint arXiv:2307.15043*, 2023.

A SCI-Defense Pseudocode

Algorithm 2 provides the complete pseudocode for SCI-Defense, with all symbolic parameters defined. Concrete values for all thresholds and weights are reported in Section 4.

Algorithm 2 SCI-Defense: Full Detection Pipeline

Require: Products $\{p_i\}$, query q, thresholds τ_s, τ_m, τ_{ppl}, weights $\lambda_{\text{AA}}, \lambda_{\text{NP}}, \lambda_{\text{CA}}, \lambda_{\text{TC}}$, boost factor β, boost thresholds $\theta_{\text{NP}}, \theta_{\text{CA}}, \theta_{\text{TC}}, \theta'_{\text{NP}}$, blend weight α

for each product p_i **do**
  $d_i \leftarrow$ concatenate(p_i.description, p_i.injected_text)

  // Step 1: Perplexity Detection (PPL)
  $\text{ppl}_i \leftarrow \exp\Big(-\frac{1}{T}\sum_{t=1}^{T}\log p_\theta(w_t \mid w_{<t})\Big)$ *// GPT-2*
  if $\text{ppl}_i > \tau_{\text{ppl}}$ **then**
    $\text{label}_i \leftarrow$ `manipulated`; **continue** *// early exit*
  end if

  // Step 2: Semantic Integrity Scoring (SIS)
  $(S_{\text{AA}}, S_{\text{NP}}, S_{\text{CA}}, S_{\text{TC}}) \leftarrow$ GPT-4o-Score(d_i) *// each* $\in [0, 1]$
  $S_{\text{base}} \leftarrow \lambda_{\text{AA}} \cdot S_{\text{AA}} + \lambda_{\text{NP}} \cdot S_{\text{NP}} + \lambda_{\text{CA}} \cdot S_{\text{CA}} + \lambda_{\text{TC}} \cdot S_{\text{TC}}$
  $\text{boost} \leftarrow (S_{\text{NP}} \geq \theta_{\text{NP}}) \vee (S_{\text{CA}} \geq \theta_{\text{CA}}) \vee (S_{\text{TC}} \geq \theta_{\text{TC}} \wedge S_{\text{NP}} \geq \theta'_{\text{NP}})$
  if boost **then**
    $S_{\text{SIS}} \leftarrow \min(S_{\text{base}} \times \beta,\ 1.0)$
  else
    $S_{\text{SIS}} \leftarrow S_{\text{base}}$
  end if

  // Step 3: Inter-Candidate Detection (ICD)
  $S_{\text{ICD}} \leftarrow$ CrossCandidateSimilarity$(p_i, \{p_j\}_{j \neq i})$
  $S_{\text{final}} \leftarrow \alpha \cdot S_{\text{SIS}} + (1 - \alpha) \cdot S_{\text{ICD}}$

  // Decision
  if $S_{\text{final}} \geq \tau_m$ **then**
    $\text{label}_i \leftarrow$ `manipulated`
  else if $S_{\text{final}} \geq \tau_s$ **then**
    $\text{label}_i \leftarrow$ `suspicious`
  else
    $\text{label}_i \leftarrow$ `clean`
  end if
end for

// Step 4: Penalization
Move all `suspicious` and `manipulated` products to last position in ranking
return defended ranking π'

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: All quantitative claims (Precision=1.000, FPR=0.000, per-attack Recall values) are directly supported by experimental results in Tables 1–6. Limitations are discussed in Section 7.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Section 7 identifies semantic relevance manipulation as a structural blind spot and discusses failure cases (five black-box attacks with Block@3=0.000). Section 5 discusses threshold calibration limitations.

Guidelines:

- The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

   Justification: This paper makes no theoretical claims requiring formal proofs; all results are empirical.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Algorithm 1 and Appendix A provide complete SCI-Defense pseudocode with all thresholds and weights. Models (GPT-2, GPT-4o) and datasets (Amazon ProductBench, MS MARCO) are publicly available. All hyperparameters are reported in Section 5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.

- If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code and data will be released upon acceptance. All datasets used (Amazon ProductBench via CORE [3], MS MARCO) are publicly available.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 5 provides full experimental setup including dataset sizes, model choices, thresholds ($\tau_{\text{ppl}} = 500$, $\tau_s = 0.55$, $\tau_m = 0.65$), SIS dimension weights, and validation split procedure for hyperparameter selection.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Experiments use deterministic components (GPT-2 perplexity, fixed SIS thresholds); LLM-based SIS scoring introduces minor variance that is not currently quantified. Confidence intervals will be reported in future work.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
   - If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: Compute resource details (hardware type, memory, API call counts) will be added in the camera-ready version.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This work studies defenses against manipulation attacks. All attack evaluations were conducted in controlled settings with no impact on real users or systems; no human subjects were involved.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Positive impact: improved integrity of LLM-based ranking systems, protecting consumers and honest sellers. Negative impact: detailed attack characterization could assist adversaries; mitigated by concurrent defense contributions and responsible release documentation.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [Yes]

    Justification: Attack code will be released with documentation emphasizing responsible use for defense research only. No high-risk pre-trained models or scraped datasets are released.

Guidelines:

- The answer [N/A] means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: GPT-4o (OpenAI API Terms of Service), GPT-2 (MIT License), MS MARCO (CC BY 4.0), and Amazon product data (publicly available) are all cited and used in accordance with their licenses.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: SCI-Defense code, evaluation datasets, and black-box attack implementations will be documented and released upon acceptance with full usage instructions.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

Justification: No human subjects were involved; all experiments were conducted on publicly available datasets.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No human subjects were involved; IRB approval was not required.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: GPT-4o is a core component of SCI-Defense's SIS scorer (Section 4.2), used to evaluate descriptions on four semantic dimensions. GPT-2 is used for perplexity-based detection (Section 4.1). Both usages are integral to the methodology.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.