Unveiling the Resilience of LLM-Enhanced Search Engines against Black-Hat SEO Manipulation

Pei Chen
peichen19@fudan.edu.cn
Fudan University
Shanghai, China

Geng Hong*
ghong@fudan.edu.cn
Fudan University
Shanghai, China

Xinyi Wu
xinyiwu20@fudan.edu.cn
Fudan University
Shanghai, China

Mengying Wu
wumy21@m.fudan.edu.cn
Fudan University
Shanghai, China

Zixuan Zhu
zhuzx24@m.fudan.edu.cn
Fudan University
Shanghai, China

Mingxuan Liu
liumx@mail.zgclab.edu.cn
Zhongguancun Laboratory
Beijing, China

Baojun Liu
lbj@tsinghua.edu.cn
Tsinghua University
Beijing, China

Mi Zhang
mi_zhang@fudan.edu.cn
Fudan University
Shanghai, China

Min Yang*
m_yang@fudan.edu.cn
Fudan University
Shanghai, China

Abstract

The emergence of Large Language Model-enhanced Search Engines (LLMSEs) has revolutionized information retrieval by integrating web-scale search capabilities with AI-powered summarization. While these systems demonstrate improved efficiency over traditional search engines, their security implications against well-established black-hat Search Engine Optimization (SEO) attacks remain unexplored.

In this paper, we present the first systematic study of SEO attacks targeting LLMSEs. Specifically, we examine ten representative LLMSE products (e.g., ChatGPT, Gemini) and construct SEO-Bench, a benchmark comprising 1,000 real-world black-hat SEO websites, to evaluate both open- and closed-source LLMSEs. Our measurements show that LLMSEs mitigate over 99.78% of traditional SEO attacks, with the phase of retrieval serving as the primary filter, intercepting the vast majority of malicious queries. We further propose and evaluate seven LLMSEO attack strategies, demonstrating that off-the-shelf LLMSEs are vulnerable to LLMSEO attacks, i.e., rewritten-query stuffing and segmented texts double the manipulation rate compared to the baseline. This work offers the first in-depth security analysis of the LLMSE ecosystem, providing practical insights for building more resilient AI-driven search systems. We have responsibly reported the identified issues to major vendors.

CCS Concepts

• **Security and privacy → Web application security**.

*Corresponding authors.

WWW '26, Dubai, United Arab Emirates.

ACM ISBN 979-8-4007-2307-0/2026/04
https://doi.org/10.1145/3774904.3792399

Keywords

LLM-Enhanced Search Engine, Search Engine Optimization, Black-Hat SEO

ACM Reference Format:
Pei Chen, Geng Hong, Xinyi Wu, Mengying Wu, Zixuan Zhu, Mingxuan Liu, Baojun Liu, Mi Zhang, and Min Yang. 2026. Unveiling the Resilience of LLM-Enhanced Search Engines against Black-Hat SEO Manipulation. In *Proceedings of the ACM Web Conference 2026 (WWW '26), April 13–17, 2026, Dubai, United Arab Emirates.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3774904.3792399

1 Introduction

Among the emerging applications of large language models (LLMs), the large language model-enhanced search engine (LLMSE) combines the vast search capabilities of the Web with efficient and precise responses to user queries. Due to their objectivity and ability to synthesize information, LLMSEs are increasingly being regarded as alternatives to traditional search engines. For instance, Perplexity, raise funds at an $18 billion valuation in early 2025 [7].

Figure 1 illustrates a comparison between LLMSE and traditional search engines. The user begins by inputting a query to a practical problem, such as "Impact of LLMSE?". The traditional search engines return several separate Web sources of information, *e.g.* news, forums. while LLMSE directly generates a well-structured response providing clearer information in a well-defined overview.

Despite these advantages, are LLMSEs truly more reliable than traditional search engines? Search engine optimization (SEO) [10], including black-hat SEO [32, 35, 43, 47, 64, 67], has damaged search result quality on the traditional search engines for decades. With many attackers now turning to LLMSEs, likely reusing established SEO methods and even inventing new manipulations, the emerging LLMSE systems are facing a qualitatively new threat, underscoring the urgency of understanding and mitigating such risks.

Research Gap. Despite the growing deployment of LLMSEs, their security under SEO manipulation remains insufficiently understood. On one hand, as a rapidly emerging field, most work on LLMSEs

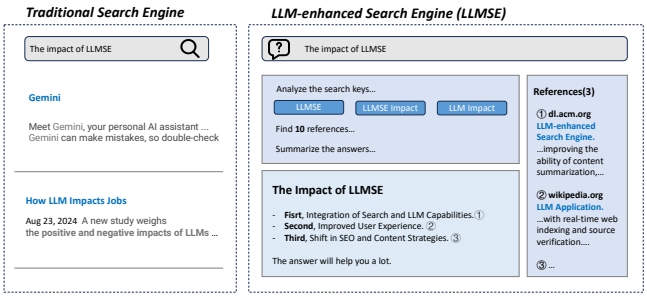

Figure 1: Examples of Traditional Search Engines vs. LLM-Enhanced Search Engines (LLMSE).

focuses on improving search efficiency, accuracy, and verifiability [39, 42, 57, 65]. On the other hand, the security-related works pay attention to the model-level attack techniques[1, 45, 54, 70]. For example, PoisonedRAG [70] manipulates RAG results by poisoning knowledge databases, and GEO [1] tries to craft text-optimization attacks to gain visibility in the model summary. However, these methods are typically evaluated in carefully crafted experimental settings, only partial components of the LLMSE workflow, neglecting their impact on the end-to-end system in real-world scenarios. Luo et al. [42] observed the harmful content and malicious URLs from LLMSE. However, they do not analyze these threats from the perspective of traditional SEO techniques or assess their phase-specific vulnerabilities. Moreover, how traditional black-hat SEO threats affect LLMSE remain underexplored, posing persistent and transferable risks to evolving LLMSE infrastructures.

Our Work. We conduct the first systematic study to investigate the black-hat SEO threat to LLMSE. We try to answer the following three important research questions (RQ):

***RQ1**: What is LLMSE workflow and whether the design of LLMSE is inherently resistant to SEO manipulations?*

***RQ2**: Will black-hat SEO attacks on traditional search engines affect LLMSE? If so, how do they affect each phase?*

***RQ3**: Are there any LLMSEO techniques that can significantly manipulating LLMSE results?*

Driven by these RQs, we first investigated 10 popular LLMSE products and analyzed the special workflow of LLMSE, revealing the attack surfaces. Second, we examined the effectiveness of traditional black-hat SEO techniques on LLMSEs. We constructed SEO-Bench with 1,000 real-world black-hat SEO attacks and then evaluated the defense performance of open-source and closed-source LLMSEs against these attacks. We further conducted a detailed empirical analysis at different phases to uncover the preferences. Finally, we propose seven LLMSEO strategies and conduct an end-to-end experiment based on the 450 self-deployment websites. All identified issues were responsibly disclosed to major LLMSE vendors.

Contribution. This work makes the following three contributions.

- We provide a detailed investigation of real-world LLMSE products, uncovering their multi-phase workflows and identifying phase-specific attack surfaces.
- We reveal that the LLMSEs can resist over 99.78% traditional black-hat SEO attacks, with the *Retrieval* phase serving as the primary filter, intercepting the vast majority of malicious queries.
- We report that LLMSEs are vulnerable to LLMSEO attacks, i.e., rewritten-query stuffing and segmented text, double the manipulation rate compared to the baseline.

2 Background

2.1 LLMSE & Black-Hat SEO

LLMSE. LLM-enhanced search engines (LLMSEs), also known as AI-powered search, combine real-time retrieval with generative summarization and are now widely adopted. Perplexity reports 169M monthly visits [56], and ChatGPT officially added search capabilities in 2024 [29]. Prior work has examined their efficiency, accuracy, and verifiability [39, 42, 57, 65], while adversarial studies explored visibility manipulation through GEO [1] and prompt injection [45, 54].

Black-Hat SEO. Search Engine Optimization (SEO) refers to improving website ranking and organic traffic through legitimate means such as optimizing structure, content, and user experience. In contrast, black-hat SEO manipulates rankings by violating search engine guidelines, aiming for short-term gains through techniques such as link farms [64], keyword stuffing [47], search redirection [32, 34], cloaking [60, 63], semantic confusion through ad injection or jargon obfuscation [36, 67], and long-tail keyword attacks [35, 37].

2.2 Threat Model

Motivated to promote specific websites, the attacker deliberately modifies the structure or content of the websites so they are favorably indexed by search engines.. When a victim user inputs certain queries, LLMSE may surface these websites and incorporate the link to the untrusted website into the generated responses.

Attacker's Goal. The attacker's objective is to induce LLMSEs to embed attacker-controlled URLs within their responses. Since link-bearing outputs directly guide users to promoted sites, our analysis focuses on responses containing clickable references to attacker-controlled domains (*e.g.* citations).

Attacker's Capability. We assumed that the attacker controls multiple websites and has full authority to customize both content and structure. However, the attacker has no access to the intermediate outputs of the LLMSE.

3 Attack Surface Analysis of LLMSE

In this section, we surveyed the current popular LLMSE systems across both open-source and closed-source markets. Through practical analysis, we can uncover the attack surface of each phase.

3.1 Representative LLMSE Collection

To gain a comprehensive understanding of the LLMSE ecosystem, we systematically collected a list of actively deployed LLMSEs from both industrial and open-source platforms. First, we searched keywords such as "LLM search" and "AI search" via Google. We extracted products from the top 100 search results and selected the five most frequently mentioned LLMSE products, which together account for over 98% of the market [58], representing those with the highest visibility and usage. Second, we surveyed popular open-source repositories in "LLM search" and selected the top five LLMSE

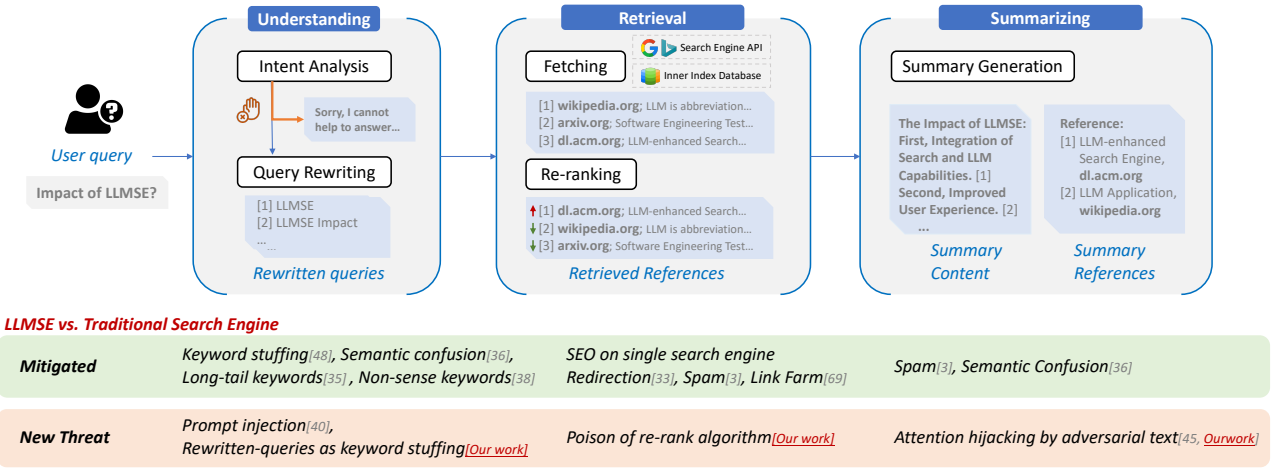

Figure 2: The Workflow and the Attack Surfaces of LLMSE. It includes three phases: (1) *Understanding*: LLMSE analyzes user query's intent and rewrites the original query; (2) *Retrieval*: fetching information from multiple databases with the rewritten queries, and re-ranking references; (3) *Summarizing*: gathers all the structured information and generates the final answer, outputting the summary content and references.

Table 1: Overview of Representative LLMSEs

LLMSE	Provider	Popularity	Pub. Date	API
Closed Source				
ChatGPT Search [50]	OpenAI	5.91B Visits	2024-10-31	Y
Gemini Grounding [20]	Google	1.06B Visits	2024-10-31	Y
Google AI Overview[22]	Google	88.5B Visits	2024-05-14	N
Perplexity [53]	Perplexity	169M Visits	2022-12-07	Y
Komo AI [28]	Komo	201k Visits	2023-01-18	Y
Open Source				
Open WebUI [9]	Open WebUI	112k Stars	2023-10-07	Y
Khoj [8]	Khoj AI	31.3k Stars	2021-04-04	N
Storm [30]	Stanford OVAL	27.5k Stars	2024-04-09	Y
Perplexica [25]	ItzCrazyKns	26.4k Stars	2024-04-09	Y
GPT Researcher [12]	Assaf Elovic	23.7k Stars	2023-05-12	Y

1) Monthly visits counts are from SimilarWeb [56]; # of stars are from Github.
2) Google AI Overview is an internal experiments module of Google Search.
3) All statistics are collected as of September 2025.

projects with over 10k stars from Github [15], reflecting strong community adoption. In total, we identified 10 representative LLMSEs, whose identities and popularity are summarized in Table 1.

3.2 Attack Surface Analysis

We conduct a comprehensive analysis on the collected LLMSE to uncover LLMSE attack surfaces. For closed-source LLMSE, we systematically reviewed their homepage descriptions and official documentation, and manually interacted with them to observe user-facing outputs. For open-source LLMSE, we deployed them locally and analyzed their outputs and server-side logs to expose intermediate processing. The workflow and the corresponding attack surface of LLMSE are summarized in Figure 2.

Phase 1: Understand the Query. LLMSE understands the user's input , and provide actionable guidance for subsequent phases. This process contains two main components:

(1.1) *Intent Analysis*: Infer user intent and decide whether external retrieval is needed (*e.g.* ChatGPT [49], Gemini [19]). This step filters irrelevant or malicious parts of the query, helping defend against attacks like irrelevant keyword stuffing and semantic confusion. However, LLM-driven inference remains susceptible to adversarial attacks such as prompt injection or jailbreaks [40].

(1.2) *Query Rewriting*: Rewrite the user input into one or more standardized queries [62]. Among the examined LLMSEs, 8/10 clearly indicate that they actively regenerate queries, often adopting a role-playing strategy [14, 26]. This step helps normalize phrasing and defend against adversarial manipulations based on typographic variations or misleading phrasing, *e.g.* the long-tail SEO [35] and non-sense keyword SEO [38]. However, it might be exploited if attackers can predict rewritten queries and tailor stuffing attacks accordingly, as further discussed in Section 5.

Phase 2: Retrieve the Information. In this phase, the LLMSE executes the rewritten queries to gather candidate retrieved references and their content for subsequent processing.

(2.1) *Fetching*: Employ external engines (*e.g.* Google [23], Tavily [59]) or inner database to fetch potentially relevant results. Some LLMSEs can restrict the search scope to curated domains [51, 52] to improve reliability. The diversity of retrieval sources helps mitigate single-source poisoning.

(2.2) *Re-ranking*: Scoring each retrieved content [19] and filter for relevance. The re-ranking process filters out spam-driven SEO abuse such as link farms, but its underlying scoring heuristics may unintentionally bias page selection, which we further examine in Section 4.4 and explore its attack implications in Section 5.

Phase 3: Summarize the Answer. After retrieval, LLMSE synthesizes the summary and typically includes in-text citations or references to enhance credibility.

(3.1) *Summary Generation*: Summarize a coherent and logically consistent response. Apart from ChatGPT and Gemini, most LLMSEs rely on external LLM APIs for content synthesis [62], with some

integrating multiple LLMs [13]. It improves factual consistency and filters out spam or semantic-confusion content, yet remains vulnerable to adversarial text that can hijack model attention [45].

4 Resilience of Black-Hat SEO Attack

Since LLMSEs are similar to traditional search engines, attackers intuitively apply existing black-hat SEO techniques to them. To assess how traditional black-hat SEO affects LLMSEs and dissect how the multi-phase mechanisms mitigate or amplify these manipulations, we conduct a comprehensive evaluation of LLMSE resilience with a large-scale real-world black-hat SEO attacks in this section.

4.1 Experiment Setup

To evaluate the resilience of LLMSEs against black-hat SEO attacks, we use Google Search as a representative traditional search engine to collect real-world successful attacks. An attack succeeding on Google but failing on an LLMSE indicates resilience. All samples from existing attacks ensure both ethical compliance and diversity.

SEO-Bench Construction. To find black-hat SEO websites, we first conducted a literature review, getting five categories of black-hat SEO attack techniques that have well-established definitions: ❶ *Semantic Confusion*: blends copied legitimate text with illicit promotions to dilute malicious intent, which raises ranking and evades filters; ❷ *Redirection*: exploits vulnerabilities on high-authority sites to forward users to promoted targets, thereby inheriting the trusted site's credibility; ❸ *Cloaking*: detects crawlers via request headers and serves SEO-optimized content to search engines while presenting different promotional or unrelated content to real users; ❹ *Keyword Stuffing*: embeds excessive or trending terms to inflate apparent relevance and manipulate ranking algorithms; ❺ *Link Farm*: creates large interlinked networks of low-quality sites to artificially raise link-based authority scores. Based on these studies, we reproduced the classification methods proposed in the corresponding works and tuned the respective classifiers. The implementation and evaluation are provided in Appendix A.

Then, we selected appropriate origin keyword queries, including illegal-words and hot-words. Illegal-words typically involve terms related to illegal or prohibited content, reflecting the underlying incentives for black-hat SEO; we extracted 2,499 illegal-words from prior studies [33, 36, 60, 67, 69]. Hot-words include popular search terms unrelated to the actual page content, which attackers use to boost visibility in rankings; we collected 9,301 hot-words from Google Trends [18] over a six-month period (Nov. 2024 – Apr. 2025). We then queried these 11,800 keywords on Google, and saved the top 50 search results, including their titles, summaries, URLs, redirection chains, and HTML content. From over 500M collected websites, we identified 1,602 valid `query-website` pairs by the classifiers. To ensure a balanced representation of each attack type in the dataset, we selected 200 pairs for each of the five SEO attack categories. As a result, our SEO-Bench dataset consists of a total of 1,000 `query-website` pairs. Table 2 shows the dataset details.

LLMSE Defense Evaluation. We evaluate nine LLMSEs introduced in Section 3, excluding Google AI Overview due to its limited and unstable availability [17]. For closed-source LLMSEs, we select their first version with full search functionalities, i.e., `gpt-4o-mini`, `gemini-1.5`, `sonar`, `komo`. For open-source LLMSEs, we deploy them locally with default configurations, and use `gpt-4o-mini` as the summarization model to ensure comparability across systems.

Table 2: Details of Black-Hat SEO Attacks in SEO-Bench

Black-Hat SEO Attack	Query	Classification Method	#
Semantic Confusion [36, 66, 67]	Illegal-words	SCDS [66]	200
Redirection [33, 34, 44, 61]	Illegal-words	Rule-Based Detector [33]	200
Cloaking [24, 46, 60, 63]	Illegal-words	Dagger [60]	200
Keywords Stuffing [3, 41, 48, 68]	Hot-words	Rule-Based Detector [48]	200
Link Farm (SSP) [6, 11, 27, 64]	Hot-words	DNS Scanner [11]	200

To quantify the resilience of LLMSEs against black-hat SEO attacks, we evaluate their ability to block target websites across different phases. Each entry in SEO-Bench is a query–website pair (q_i, t_i), where q_i is a search query and t_i is the associated black-hat SEO website. We independently evaluate the resilience of each phase using a phase-specific blocking rate, defined as the proportion of attacks intercepted at that phase among those entering the phase. Specifically: In *Understanding* phase, an attack is blocked if the LLMSE decides not to proceed with retrieval after analyzing, indicating an early rejection of the search. In *Retrieval* phase, an attack is blocked if the LLMSE performs retrieval but the SEO website does not appear in the retrieved results. In *Summarizing* phase, an attack is blocked if the SEO website appears in the retrieval reference but is excluded from the final reference. We also employ *Cumulative Resilience* to intuitively capture the overall interception achieved after each phase. The metrics are in Appendix B.

4.2 Landscape

We assess nine LLMSEs with three trials per query (27,000 requests). Results are summarized in Table 3.

Our evaluation shows that LLMSEs are highly effective against black-hat SEO attacks, with the *Understanding*, *Retrieval*, and *Summarizing* phases blocking 15.7%, 98.2%, and 85.2% of attacks at their respective phases. Overall, they achieve a cumulative blocking rate of 99.78%, where *Retrieval* plays the most decisive role by filtering the majority, and *Summarizing* adds a strong safeguard before output generation. These results highlight the importance of layered defenses in LLM-based systems, enabling them to significantly outperform traditional search engines in resisting SEO attacks.

Although the results show that black-hat SEO techniques can still influence LLMSEs, the resilience varies significantly across LLMSEs. ChatGPT is not affected by any black-hat SEO attack with a high refusal rate. Notably, open-source LLMSEs keep great defense performance, due to the fact that they choose the search engine API such as Tavily [59] or SearXNG [55], which provide optimized source. In contrast, Komo and Perplexity are most affected.

The impact of different types of black-hat SEO attacks on LLMSEs varies as well. Semantic Confusion and cloaking pose the greatest risks in the final output to LLMSEs. For example, Komo is severely affected by Semantic Confusion, with a low filtering of 83.0% and 73.5%. Meanwhile, Gemini is only affected by Semantic Confusion. In contrast, although both redirection and cloaking attacks have successfully passed the *Retrieval* phase on some LLMSEs, few of them advanced to the *Summarizing* phase, thus failing to achieve a successful attack on the *Summarizing* phase. Besides, the Keyword Stuffing poses no influence on any LLMSE.

Table 3: Performance of Black-Hat SEO Attacks on LLMSEs. Each item: Resilience (Und) / Resilience (Ret) / Resilience (Sum).

LLMSE	Total Performance	Semantic Confusion	Redirection	Cloaking	Keywords Stuffing	Link Farm
ChatGPT	75.8% / - / 100%	92.5% / - / 100%	79.0% / - / 100%	84.0% / - / 100%	62.0% / - / 100%	61.5% / - / 100%
Gemini	16.3% / - / 99.8%	38.0% / - / 99.0%	4.5% / - / 100%	24.0% / - / 100%	3.0% / - / 100%	12.0% / - / 100%
Perplexity	4.6% / 98.2% / 64.7%	12.0% / 94.9% / 55.6%	1.5% / 99.0% / 50.0%	6.0% / 97.9% / 100%	0 / 100% / -	3.5% / 99.0% / 50.0%
Komo	4.2% / 94.9% / 67.3%	0 / 83.0% / 73.5%	1.0% / 99.0% / 100%	20.0% / 93.8% / 50.0%	0 / 100% / -	0 / 98.5% / 33.3%
Open-WebUI	8.1% / 96.0% / 100%	12.1% / 93.1% / 100%	0 / 100% / -	7.7% / 91.7% / 100%	0 / 100% / -	17.9% / 95.7% / 100%
Khoj	30.9% / 99.6% / 100%	61.0% / 100% / -	19.5% / 100% / -	36.5% / 100% / -	16.5% / 100% / -	21.0% / 98.1% / 100%
Storm	0 / 99.8% / 50.0%	0 / 100% / -	0 / 100% / -	0 / 100% / -	0 / 100% / -	0 / 99.0% / 50.0%
Perplexica	0 / 100% / -	0 / 100% / -	0 / 100% / -	0 / 100% / -	0 / 100% / -	0 / 100% / -
GPT Researcher	1.4% / 95.5% / 100%	0 / 88.5% / 100%	0 / 97.5% / 100%	0 / 95.0% / 100%	3.0% / 100% / -	3.9% / 98.0% / 100%
Average Res.	15.7% / 98.2% / 85.2%	24.0% / 95.4% / 88.0%	11.7% / 99.5% / 90.0%	19.8% / 97.6% / 91.7%	9.4% / 100.0% / 100.0%	13.3% / 98.7% / 79.2%
Cumulative Res.	15.7% / 98.48% / 99.78%	24.0% / 96.50% / 99.58%	11.7% / 99.56% / 99.96%	19.8% / 98.08% / 99.84%	9.4% / 100.00% / 100.00%	13.3% / 99.00% / 99.96%

► **Finding I**: *LLMSEs exhibit strong resilience to black-hat SEO attacks, achieving the cumulative blocking rate of 99.78%. Retrieval phase intercepts the vast majority of malicious queries. Semantic Confusion poses the greatest risks to LLMSEs.*

4.3 Resilience on Understanding Phase

Firstly, we measure how LLMSEs interpret and rewrite the query in the *Understanding* phase helps mitigate black-hat SEO attacks.

Intent Interception. Upon receiving a user query, LLMSEs infer its intent to decide whether a web search is necessary. We analyze the interception mechanism by inspecting the specific API field values, as ChatGPT and Gemini explicitly indicate whether "web_search" is invoked. Then we manually check these refused queries and corresponding answers. In our result, 75.8% of queries on ChatGPT are intercepted with no reference. Among them, 30.6% are refused due to violations of safety policies, and 69.4% are skipped due to intent misinterpretation, where the system treats the input as a statement rather than a search query. This is because, unlike dedicated search engines, LLMSEs such as ChatGPT, which treat search as an auxiliary function, tend not to invoke search when they can answer based on internal knowledge. Similarly, Gemini intercepts 16.3% of queries, with 67.4% refusals and 32.6% misinterpretations. Although intent interception is not designed to counter SEO attacks, it can incidentally filter harmful or illicit queries before search execution, thereby reducing exposure to malicious content.

► **Finding II**: *Intent interception enables LLMSEs such as ChatGPT to filter out 75.8% of queries, effectively disrupting malicious SEO attempts at the start, even though unintentionally.*

Query Rewriting. Before performing a real search in the *Retrieval* phase, some LLMSEs generate a refined version of the original query. To investigate how this step influences the effectiveness of SEO-based manipulations, we extract the rewritten keywords on five of the collected LLMSEs that provide the rewritten queries in their API response, and compare them with the original inputs. Our analysis reveals that almost all of them prefer to rewrite the query. Figure 3 shows the changes in word count during the rewriting. For example, GPT Researcher shows a strong tendency to expand the query (77.56%), while others are more likely to shorten it. Further examination of the system prompts indicates that the rewriting mechanisms, such as prefix/suffix modifications, query formatting, and targeted semantic enrichment, are guided by instructions aimed at improving search accuracy and user experience.

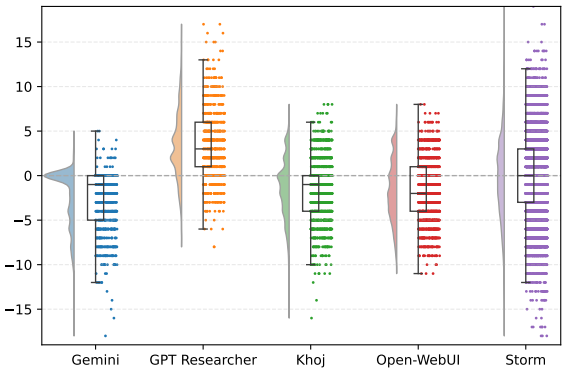

Figure 3: Distribution of Word Counts after Rewriting.

To further examine how rewriting can mitigate the influence of black-hat SEO, we conducted a supplementary validation experiment on query rewriting. As detailed in Appendix C, reissuing rewritten queries to Google Search showed that 98.16% failed to retrieve the original SEO websites, with even minor edits (edit distance below 0.1) reducing attack success rates to under 10%, confirming the strong disruptive impact of rewriting on adversarial rankings. These results indicate that even slight syntactic modifications can substantially suppress exposure, suggesting that query rewriting serves as an effective and lightweight countermeasure.

► **Finding III**: *Query rewriting by LLMSEs effectively disrupts SEO attacks, including Long-tail and Keyword Stuffing. Even small edits can reduce the attack success rate to under 10%.*

4.4 Resilience on Retrieval Phase

Then, we measure how fetching and re-ranking in the *Retrieval* phase helps mitigate black-hat SEO attacks.

Fetching Preferences. LLMSEs can restrict the search scope during fetching, so we examine whether their returned links are of higher quality than those from traditional engines. Using domain rankings [31], we analyze retrieved references collected from

trending queries. As shown in Figure 4, most LLMSEs favor higher-ranked domains, with top-5k links appearing more frequently than lower-ranked ones. For instance, ChatGPT's share of authoritative links is nearly 50% higher than Google's. In contrast, LLMSEs such as Khoj and Perplexica exhibit over 60% unreliable links due to hallucinated URLs. However, while prioritizing authoritative sources improves result reliability and reduces SEO risks, excessive trust in high-ranking domains may introduce new vulnerabilities, such as malicious redirects or comment-based attacks on reputable sites.

▸ **Finding IV**: *LLMSE fetching preferences for authoritative websites enhance the overall search quality, but also emphasize the risk of compromised high-ranking sources.*

Re-ranking Preferences. After fetching the content of web pages, LLMSEs often re-rank them based on internally defined quality assessment criteria. To investigate this preference, we examine the rank shifts of links that appear in both Google search results and LLMSEs. For each link that occurs in both result sets, we compute its relative rank change within the respective systems as

$$\Delta RelRank_i = Rank^i_{LLM} - Rank^i_{Google} \quad (1)$$

where $Rank^i_{Google}$ denotes the relative rank of the i-th link within the set of overlapping links in the Google search results, and $Rank^i_{LLM}$ denotes its relative rank in the LLMSEs results. To further uncover the re-ranking criteria employed by LLMSEs, we categorize websites with increasing relative rankings (i.e., $\Delta RelRank_i < 0$) as *up* sites, and those with decreasing relative ranks (i.e., $\Delta RelRank_i > 0$) as *down* sites. Then, we analyze several common features of the websites by computing the average values and their corresponding rates of differences, as summarized in Table 4.

The results indicate that websites with increased rankings in LLMSEs typically exhibit a higher degree of text fragmentation (+19.04%) and a greater presence of multimodal resources (+18.71%) in terms of content. Structurally, they tend to feature denser internal linking (+14.89%) and greater DOM depth (+11.36%), which reflects a higher level of formatting complexity. These observed differences suggest a set of potentially influential factors that may reflect the preferences of LLMSEs. We will further examine their actual impact through controlled experiments in Section 5.

▸ **Finding V**: *During the re-ranking, LLMSEs tend to favor webpages with higher content quality and content richness.*

4.5 Resilience on Summarizing Phase

Finally, we measure summary intercepting of *Summarizing* phase.

Summary Interception. To evaluate LLMSE resilience against illicit promotion during summary generation, we examine their ability to filter malicious content from illegal queries. We use an illicit-website classifier (Appendix A) to measure the proportion of illegal references in *Retrieval* and *Summary* phases. Additionally, we analyze the semantics of the responses to determine how many illicit links contaminate the generated summaries. As shown in Table 5, LLMSEs increasingly block malicious content as generation proceeds. In the *Summary* phase, they remove on average 75.3% more illegal links than in the *Retrieval* phase, with Open-WebUI achieving the largest reduction (97.9%). Furthermore, 70.8% illicit content is intercepted from the summary reference to the answer content. These results indicate that LLMSEs generally favor neutral or positive content and actively suppress outputs involving violence, pornography, or other harmful material, reflecting their built-in safety mechanisms and alignment with normative standards.

We further analyze intercepted SEO attacks in the *Summarizing* phase, including *Semantic Confusion* and *Cloaking*. Such pages often contain irrelevant or mixed content, weakening semantic relevance to the query. For example, confusion attacks may embed illicit promotions within otherwise legitimate text to enhance credibility. Appearing benign, they diverge from the illegal intent and are excluded from summaries. This suggests that, beyond rejecting

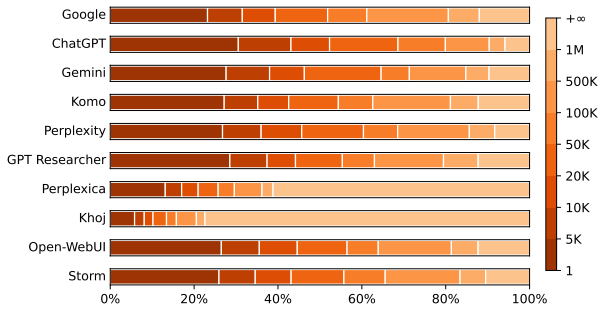

Figure 4: Domain Rank Proportions of Retrieved References from Both Traditional Search Engines and LLMSEs.

Table 4: Different Features of Re-Ranked Websites

Features	Avg. (*Up*)	Avg. (*Down*)	Differences	P-value
Text Fragmentation	60.09	50.48	**+19.04%**	**0.0100**
DOM Depth	13.93	12.51	**+11.36%**	**0.0036**
Tag Diversity	22.61	22.27	+1.543%	0.8674
External Link #	14.71	15.81	-6.971%	0.9901
Internal Link #	45.66	39.74	**+14.89%**	**0.0003**
Multi-modal #	12.50	10.53	**+18.71%**	**0.0292**
Meta Completeness	0.4167	0.4196	-0.6911%	0.8809
Alt Coverage	0.2899	0.2873	+0.9050%	0.8540

If p-value < 0.05, the difference is considered statistically significant [4].

Table 5: Illegal Proportion in Different Phases.

LLMSE	Retrieved References	Summary References (δ)	Summary Content (δ)
ChatGPT	–	2.00% (–)	2.00% (0.0%)
Gemini	–	0.00% (–)	0.00% (0.0%)
Komo	4.36%	1.53% (↓ 64.9%)	0.28% (↓ 81.7%)
Perplexity	0.69%	0.35% (↓ 49.3%)	0.00% (↓ 100.0%)
GPT Researcher	10.07%	3.48% (↓ 65.4%)	1.55% (↓ 55.5%)
Perplexica	1.24%	0.19% (↓ 84.7%)	0.00% (↓ 100.0%)
Khoj	1.27%	0.05% (↓ 96.1%)	0.00% (↓ 100.0%)
Open-WebUI	10.98%	0.23% (↓ 97.9%)	0.00% (↓ 100.0%)
Storm	1.61%	0.50% (↓ 69.0%)	0.00% (↓ 100.0%)
Avg.	4.89%	0.92% (↓ 75.3%)	0.43% (↓ 70.8%)

harmful content, LLMSEs prioritize sources semantically aligned with query intent, which further mitigates SEO attacks.

> ► **Finding VI**: *LLMSEs prefer benign and semantically aligned content in summarization, refusing additional 75.3% illegal websites. This strategy further mitigates the threat of illicit content and the impact of attacks such as Semantic Confusion.*

5 LLMSEO Attack

Building on these findings, we propose and evaluate novel end-to-end attacks tailored to LLMSEs, extending beyond prior work that focused only on ranking or summarization components [1, 2, 70].

5.1 LLMSEO Attack Techniques

As each phase of the LLMSE workflow (*Understanding*, *Retrieval*, and *Summarizing*) exposes distinct security risks, we design phase-specific LLMSEO attack strategies to systematically evaluate and manipulate these vulnerabilities.

Attacking Understanding Phase. In this phase, rewriting introduces ambiguity and susceptibility to manipulation. To exploit this vulnerability, we design a targeted attack.

- *Rewritten-query Stuffing.* Embed potential rewritten queries extensively within web pages. Since most LLMSE may not always use the original search key for retrieval (Section 3), predicting possible new queries and inserting them in web content can increase the likelihood of being matched.

Note that while prompt injection is an important attack vector in this phase (see Section 3), its impact is well studied [5, 40, 45, 54] and highly prompt-dependent, so we exclude it from our experiments.

Attacking Retrieval Phase. Aiming to increase the priority in re-ranking and inspired by Finding V, we propose new techniques designed to align with the scoring mechanism.

- *Internal Links.* Embed numerous internal links within web pages to construct a network, simultaneously increasing the number of links to key pages.
- *Multi-modal Resources.* Incorporate multi-modal resources (*e.g.* text, images, and videos) into web pages to increase their richness and boost perceived credibility.
- *Nested Structure.* Use structured labels and indexing that are better suited for retrieval to enhance the readability and accessibility of content, helping LLMSE quickly locate and extract key information, gaining an advantage in candidate selection.
- *Segmented Text.* Reduce the length of individual text segments. Shorter texts are often more suitable for direct citation.

Attacking Summarizing Phase. To influence this phase, we design optimization strategies targeting both relevance and format.

- *Relevance Enhancement.* Focus on core keywords to enhance the semantic relevance and coherence of the text to the query.
- *Q&A Formatting.* Present content in a question-and-answer (Q&A) format in the conversational tone of LLM-generated responses, increasing the likelihood of direct reuse by LLMSE.

5.2 Effectiveness Evaluation

To examine the applicability of these LLMSE attack techniques influencing LLMSEs, we conducted a competitive experiment to compare the efficacy of various attack strategies under real-world conditions.

Methodology. To evaluate the effectiveness of different LLMSE attack techniques, we deployed blog websites under a controlled domain, each promoting a different brand of the same type of product. We then queried LLMSEs for product recommendations using domain-restricted prompts (Appendix D), and recorded the proportion of recommended sites associated with each attack technique. This restriction reduces ethical concerns arising from real-world search pollution while enabling fair comparison across techniques. A higher occurrence suggests a stronger alignment between the corresponding manipulation and the preferences of the LLMSE.

To mitigate the impact of randomness in LLMSE responses, the query was repeated 10 times per LLMSE, and we aggregated the total number of times each site appeared in the responses. In addition to the seven LLMSE attack types, we included one non-SEO and one traditional SEO attack (*i.e.* Semantic confusion, which showed the best performance in Section 4.2) for baseline comparison. For each attack type, 50 websites were created. In total, the experiment involved 450 adversarial websites. To ensure ethical compliance, all websites were labeled as "For Testing Purposes Only" and taken offline after the experiment finished. This ensured minimal long-term impact while maintaining the integrity of real-world testing.

Implementation. We implemented these LLMSE attacks across various websites. Specifically, we first generated a set of products using the same pattern, *i.e.* "Brand" + "Entity" noun, and generated the base content with `gpt-4o-mini`. We also use the model for *Rewritten-query Stuffing*, generating rewritten queries and embedding them into web pages. For the *Internal Links*, we embedded hyperlinks among the 50 websites of this type, forming mutual linking connections in the "Useful Links" block. For the *Multi-modal Resources*, we doubled the number of images in the base website content to increase visual richness. In the *Nested Structure*, we added an additional layer of subheadings, expanding from second-level to third-level headers to increase structural complexity. For the *Segmented Text*, we restructured the content by halving the average paragraph length, resulting in more segmented text blocks. For the *Relevance Enhancement*, we removed the irrelevant part, *e.g.* teams, and added more descriptions about products. In the *Q&A Formatting*, each paragraph was prefaced with a question, followed by a corresponding answer block to simulate a question-answer format. In the *Semantic Confusion*, we inserted a promotional segment into unmodified news content.

Results. Table 6 presents the performance of LLMSEO attacks on LLMSEs, where each row shows the proportion of a specific attack type among all successful attacks. [1] Overall, all proposed attacks demonstrated measurable effectiveness on LLMSEs, with each achieving performance above the baseline in at least one LLMSE. In all attacks, attacks targeting the *Retrieval* phase were more effective. *Segmented Text* achieved the highest attack across most LLMSE platforms, with an exposure rate exceeding 50% on Perplexica and GPT-Researcher, indicating that LLMSEs are better at understanding short and segmented content. The second most effective technique was *Rewritten-query Stuffing*, which doubled the

[1] Notably, `gpt-4o-mini` and `gemini-1.5` refused to access all provided URLs in this experiment, likely due to stricter content-fetching policies [16].

Table 6: Success Rate of LLMSEO Attacks on LLMSEs. The percentage in each row indicates the success rate of a specific approach among all successful attacks. The bold numbers highlight the most effective attacks, and the up arrows (↑) indicate the most improving attacks between *Summarizing* phase and *Retrieval* phase.

LLMSE	Exposed Phase	Baseline		Understanding	Retrieval				Summarizing	
		Blank	Semantic Confusion	Rewritten-query Stuffing	Internal Links	Multi-modal Resources	Nested Structure	Segmented Text	Relevance Enhancement	Q&A Formatting
Perplexity	Retrieval	7.29%	0.00%	19.79%	10.42%	8.33%	6.25%	**28.12%**	11.46%	8.33%
	Summarizing	0.00%	0.00%	25.00%	12.50%	12.50%	0.00%	**37.50%** ↑	12.50%	0.00%
Komo	Retrieval	8.33%	0.00%	23.96%	10.42%	9.38%	1.04%	**30.21%**	10.42%	6.25%
	Summarizing	2.44%	0.00%	26.83%	2.44%	4.88%	0.00%	**41.46%** ↑	14.63%	7.32%
Open-WebUI	Retrieval	9.09%	2.60%	27.27%	2.60%	3.90%	11.69%	**28.57%**	0.00%	14.29%
	Summarizing	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%	0.00%
Khoj	Retrieval	12.38%	3.47%	16.34%	4.95%	6.93%	**26.73%**	17.33%	4.46%	7.43%
	Summarizing	15.29%	1.18%	16.47%	4.71%	2.35%	**30.59%**	22.35% ↑	1.18%	5.88%
Storm	Retrieval	13.9%	1.36%	**23.71%**	10.08%	9.26%	15.80%	9.54%	4.90%	11.44%
	Summarizing	14.03%	0.45%	16.29%	10.86%	11.31%	**19.91%** ↑	8.14%	5.88%	13.12%
Perplexica	Retrieval	20.25%	0.00%	7.59%	5.06%	3.8%	7.59%	**45.57%**	10.13%	0.00%
	Summarizing	19.35%	0.00%	0.00%	6.45%	3.23%	6.45%	**51.61%** ↑	12.9%	0.00%
GPT-Researcher	Retrieval	0.00%	0.00%	21.74%	2.17%	6.52%	0.00%	**65.22%**	0.00%	4.35%
	Summarizing	0.00%	0.00%	24.32% ↑	2.70%	8.11%	0.00%	**59.46%**	0.00%	5.41%

exposure rate compared to the baseline in five LLMSEs, highlighting its strong influence on downstream outcomes.

We further compare attack proportions across phases to assess each phase's filtering effects. As *Summarizing* proportions post-*Retrieval* filtering, we focus on the attack effectiveness differences between the two phases to capture summary interception. In Table 6, we mark significant increases in attack proportions during *Summarizing*. Content-driven strategies, such as *Segmented Text* and *Relevance Enhancement*, show noticeable growth, while *Semantic Confusion*, as well as link- and resource-based tactics (*Internal Links*, *Multi-modal Resources*), tend to decline. This shift suggests that, in *Summarizing* phase, LLMSE is more influenced by content quality rather than the structure of external resources, underlining the importance of content-level manipulation.

6 Discussion

Security Implications. The application of LLMSEs in information retrieval reshapes user trust and the threat landscape of black-hat SEO. Users over-rely on LLMSE-generated summaries and references, perceiving them as authoritative, which magnifies the risks when malicious content and links are included in trusted outputs. Building upon traditional black-hat SEO, attackers are increasingly adapting their strategies to the internal preferences of LLMSEs, shifting content manipulation from isolated optimizations to system-level adversarial interactions. As LLMSE adoption grows, such weaknesses may gradually distort the Web's credibility structure, highlighting the need for timely, robust defenses to support a healthy information ecosystem and sustained user trust.

Mitigation. To mitigate the vulnerabilities identified in this study, defenses for LLMSEs should be reinforced in a phase-aware manner across the entire workflow. In the *Understanding* phase, analyzing the stability of query rewriting and intent interpretation can help detect inputs deliberately aligned with rewriting behaviors; paraphrasing-based filtering, as shown in PoisonedRAG [70], can counter poisoning attempts in RAG systems. During *Retrieval*, mitigation should go beyond static domain authority by incorporating behavior-based signals, such as redirection patterns and cross-query reference consistency, to identify abused high-credibility sources. In the *Summarizing* phase, additional safeguards are needed against prompt- or text-based manipulations embedded in retrieved pages that may bias summarization preferences. Furthermore, user awareness and transparency features, such as link provenance or credibility indicators, are crucial to reduce over-reliance on generated outputs and promote critical content verification, collectively enhancing the resilience of the LLMSE ecosystem.

Limitation. Our evaluation focused on ten representative LLMSEs selected by user scale and popularity, though other systems beyond this scope may demonstrate stronger resilience. For ethical reasons, our implemented LLMSEO attacks were intentionally simplified and deployed for limited durations; real-world adversaries may employ more sophisticated or persistent methods, and the combined effects of multiple strategies remain unexplored.

7 Conclusion

This work presents the first systematic security analysis of Large Language Model-enhanced Search Engines (LLMSEs), revealing how black-hat SEO continues to influence their behaviors. By analyzing phase-specific preferences and weaknesses, we demonstrate effective LLMSEO attacks that exploit these vulnerabilities. We offer insights into more secure and resilient AI-driven search systems.

Acknowledgments

This work was supported by the New Generation Artificial Intelligence-National Science and Technology Major Project (No. 2025ZD0123204). Min Yang is a faculty of Shanghai Pudong Research Institute of Cryptology, Shanghai Institute of Intelligent Electronics & Systems and Engineering Research Center of Cyber Security Auditing and Monitoring, and Shanghai Collaborative Innovation Center of Intelligent Visual Computing, Ministry of Education, China.

References

[1] Pranjal Aggarwal, Vishvak Murahari, Tanmay Rajpurohit, Ashwin Kalyan, Karthik Narasimhan, and Ameet Deshpande. 2024. Geo: Generative engine optimization. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, New York, NY, USA, 5–16.

[2] Marwah Alaofi, Paul Thomas, Falk Scholer, and Mark Sanderson. 2024. LLMs can be Fooled into Labelling a Document as Relevant: best café near me; this paper is perfectly relevant. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region (SIGIR-AP '24)* (Tokyo, Japan) *(SIGIR-AP 2024)*. Association for Computing Machinery, New York, NY, USA, 32–41. doi:10.1145/3673791.3698431

[3] Lourdes Araujo and Juan Martinez-Romo. 2010. Web spam detection: new classification features based on qualified link analysis and language models. *IEEE Transactions on Information Forensics and Security* 5, 3 (2010), 581–590.

[4] Peter Bruce and Andrew Bruce. 2017. *Practical Statistics for Data Scientists*. O'Reilly Media, Sebastopol, CA, USA.

[5] Sizhe Chen, Arman Zharmagambetov, Saeed Mahloujifar, Kamalika Chaudhuri, David Wagner, and Chuan Guo. 2025. SecAlign: Defending Against Prompt Injection with Preference Optimization. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*. ACM, Taipei. Preprint at arXiv:2410.05451.

[6] Young-joo Chung, Masashi Toyoda, and Masaru Kitsuregawa. 2009. A study of link farm distribution and evolution using a time series of web snapshots. In *Proceedings of the 5th international workshop on Adversarial information retrieval on the Web*. ACM, New York, NY, USA, 9–16.

[7] CNBC. 2025. AI startup Perplexity in talks to double valuation to $1.8 billion via new funding. https://www.cnbc.com/2025/03/20/perplexity-in-talks-to-double-valuation-to-18-billion-via-new-funding.html. Accessed: 2025-04-19.

[8] Khoj AI Contributors. 2023. Khoj: Natural language second brain. https://github.com/khoj-ai/khoj Accessed: 2025-04-10.

[9] Open WebUI Contributors. 2023. Open WebUI: A self-hosted ChatGPT UI. https://github.com/open-webui/open-webui Accessed: 2025-04-10.

[10] Wikipedia contributors. 2025. Search engine optimization. https://en.wikipedia.org/wiki/Search_engine_optimization Accessed: 2025-01-22.

[11] Kun Du, Hao Yang, Zhou Li, Haixin Duan, and Kehuan Zhang. 2016. The ever-changing labyrinth: a large-scale analysis of wildcard DNS powered blackhat SEO. In *Proceedings of the 25th USENIX Conference on Security Symposium (SEC'16)*. USENIX Association, Austin, TX, USA, 245–262.

[12] Assaf Elovic. 2023. GPT Researcher: Autonomous agent for comprehensive online research. https://github.com/assafelovic/gpt-researcher Accessed: 2025-04-10.

[13] Assaf Elovic. 2023. gpt-researcher Documentation. https://gptr.dev/#features. Accessed: 2025-04-20.

[14] Assaf Elovic. 2023. gpt-researcher Prompt Example. https://github.com/assafelovic/gpt-researcher/blob/dbc0bbe4cf55d918511b779cc320454b153415f1/gpt_researcher/prompts.py#L15. Accessed: 2025-04-20.

[15] felladrin. 2023. Awesome AI Web Search. https://github.com/felladrin/awesome-ai-web-search. Accessed: 2025-04-07.

[16] Google. 2024. Gemini API: URL Context. https://ai.google.dev/gemini-api/docs/url-context. Accessed: 2025-06-06.

[17] Google. 2024. How AI Overview works in Google Search. https://support.google.com/websearch/answer/14901683. Accessed: 2025-04-07.

[18] Google. 2025. Hot Trends. http://www.google.com/trends/hottrends Accessed: 2025-01-01.

[19] Google AI. 2024. Gemini API Grounding Documentation. https://ai.google.dev/gemini-api/docs/grounding. Accessed: 2025-04-20.

[20] Google AI Team. 2025. Gemini by Google. https://gemini.google.com. Accessed: 2025-01-11.

[21] Google Developers. 2024. Google crawlers: User agents used by Googlebot. https://developers.google.com/search/docs/crawling-indexing/google-common-crawlers Accessed: 2025-04-24.

[22] Google Developers. 2025. AI Overviews and Your Website. https://developers.google.com/search/docs/appearance/ai-overviews. Accessed: 2025-01-11.

[23] Google Developers. 2025. Custom Search JSON API. https://developers.google.com/custom-search/v1/overview. Accessed: 2025-01-11.

[24] Luca Invernizzi, Kurt Thomas, Alexandros Kapravelos, Oxana Comanescu, Jean-Michel Picod, and Elie Bursztein. 2016. Cloak of Visibility: Detecting When Machines Browse a Different Web. In *2016 IEEE Symposium on Security and Privacy (SP)*. IEEE, Piscataway, NJ, USA, 743–758. doi:10.1109/SP.2016.50

[25] ItzCrazyKns. 2023. Perplexica: Self-hosted AI-powered search engine. https://github.com/ItzCrazyKns/Perplexica Accessed: 2025-04-10.

[26] ItzCrazyKns. 2024. Perplexica WebSearch Prompt Code. https://github.com/ItzCrazyKns/Perplexica/blob/master/src/lib/prompts/webSearch.ts#L1. Accessed: 2025-04-20.

[27] John P John, Fang Yu, Yinglian Xie, Arvind Krishnamurthy, and Martin Abadi. 2011. {deSEO}: Combating {Search-Result} Poisoning. In *20th USENIX Security Symposium (USENIX Security 11)*. USENIX Association, Berkeley, CA, USA.

[28] Komo. 2025. Komo AI. https://komo.ai/. Accessed: 2025-04-07.

[29] Reinhardt Krause. 2024. Google Stock Falls on OpenAI's SearchGPT Debut. Investor's Business Daily. https://www.investors.com/news/technology/google-stock-tumbles-on-openais-searchgpt-debut/

[30] Stanford OVAL Lab. 2024. Storm: Structured LLMs with Open-domain Retrieval and Memory. https://github.com/stanford-oval/storm Accessed: 2025-04-10.

[31] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczyński, and Wouter Joosen. 2019. Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. In *Proceedings of the 26th Annual Network and Distributed System Security Symposium (NDSS 2019)*. The Internet Society, Reston, VA, USA. doi:10.14722/ndss.2019.23386

[32] Nektarios Leontiadis, Tyler Moore, and Nicolas Christin. 2011. Measuring and Analyzing Search-Redirection Attacks in the Illicit Online Prescription Drug Trade. In *Proceedings of the 20th USENIX Conference on Security* (San Francisco, CA) *(SEC'11)*. USENIX Association, USA, 19.

[33] Nektarios Leontiadis, Tyler Moore, and Nicolas Christin. 2011. Measuring and analyzing {Search-Redirection} attacks in the illicit online prescription drug trade. In *20th USENIX Security Symposium (USENIX Security 11)*. USENIX Association, Berkeley, CA, USA.

[34] Nektarios Leontiadis, Tyler Moore, and Nicolas Christin. 2014. A Nearly Four-Year Longitudinal Study of Search-Engine Poisoning. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security* (Scottsdale, Arizona, USA) *(CCS '14)*. Association for Computing Machinery, New York, NY, USA, 930–941. doi:10.1145/2660267.2660332

[35] Xiaojing Liao, Chang Liu, Damon McCoy, Elaine Shi, Shuang Hao, and Raheem Beyah. 2016. Characterizing long-tail SEO spam on cloud web hosting services. In *Proceedings of the 25th International Conference on World Wide Web* (Montréal, Québec, Canada) *(WWW '16)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 321–332.

[36] Xiaojing Liao, Kan Yuan, XiaoFeng Wang, Zhongyu Pei, Hao Yang, Jianjun Chen, Haixin Duan, Kun Du, Eihal Alowaisheq, Sumayah Alrwais, et al. 2016. Seeking nonsense, looking for trouble: Efficient promotional-infection detection through semantic inconsistency search. In *2016 IEEE Symposium on Security and Privacy (SP)*. IEEE, 707–723.

[37] Jiakun Liu, Sebastian Baltes, Christoph Treude, David Lo, Yun Zhang, and Xin Xia. 2021. Characterizing search activities on stack overflow. In *Proceedings of the 29th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 919–931.

[38] Mingxuan Liu, Yunyi Zhang, Lijie Wi, Baojun Liu, Geng Hong, Yiming Zhang, Hui Jiang, Jia Zhang, Haixin Duan, Min Zhang, Wei Guan, Fan Shi, and Min Yang. 2025. NOKEScam: understanding and rectifying non-sense keywords spear scam in search engines. In *Proceedings of the 34th USENIX Conference on Security Symposium* (Seattle, WA, USA) *(SEC '25)*. USENIX Association, USA, Article 246, 20 pages.

[39] Nelson F. Liu, Tianyi Zhang, and Percy Liang. 2023. Evaluating Verifiability in Generative Search Engines. arXiv:2304.09848 [cs.CL] https://arxiv.org/abs/2304.09848

[40] Yupei Liu, Yuqi Jia, Runpeng Geng, Jinyuan Jia, and Neil Zhenqiang Gong. 2024. Formalizing and benchmarking prompt injection attacks and defenses. In *33rd USENIX Security Symposium (USENIX Security 24)*. 1831–1847.

[41] Long Lu, Roberto Perdisci, and Wenke Lee. 2011. Surf: detecting and measuring search poisoning. In *Proceedings of the 18th ACM conference on Computer and communications security*. 467–476.

[42] Zeren Luo, Zifan Peng, Yule Liu, Zhen Sun, Mingchen Li, Jingyi Zheng, and Xinlei He. 2025. Unsafe LLM-based search: quantitative analysis and mitigation of safety risks in AI web search. In *Proceedings of the 34th USENIX Conference on Security Symposium* (Seattle, WA, USA) *(SEC '25)*. USENIX Association, USA, Article 413, 20 pages.

[43] Richard McCreadie, Craig Macdonald, Iadh Ounis, Jim Giles, and Ferris Jabr. 2012. An examination of content farms in web search using crowdsourcing. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management* (Maui, Hawaii, USA) *(CIKM '12)*. Association for Computing Machinery, New York, NY, USA, 2551–2554. doi:10.1145/2396761.2398689

[44] Hesham Mekky, Ruben Torres, Zhi-Li Zhang, Sabyasachi Saha, and Antonio Nucci. 2014. Detecting malicious HTTP redirections using trees of user browsing activity. In *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*. 1159–1167. doi:10.1109/INFOCOM.2014.6848047

[45] Fredrik Nestaas, Edoardo Debenedetti, and Florian Tramèr. 2024. Adversarial Search Engine Optimization for Large Language Models. arXiv:2406.18382 [cs.CR] https://arxiv.org/abs/2406.18382

[46] Yuan Niu, Yi-Min Wang, Hao Chen, Ming Ma, and Francis Hsu. 2006. *A Quantitative Study of Forum Spamming Using Context-based Analysis*. Technical Report MSR-TR-2006-173. 14 pages. https://www.microsoft.com/en-us/research/publication/a-quantitative-study-of-forum-spamming-using-context-based-analysis/

[47] Alexandros Ntoulas, Marc Najork, Mark Manasse, and Dennis Fetterly. 2006. Detecting spam web pages through content analysis. In *Proceedings of the 15th International Conference on World Wide Web* (Edinburgh, Scotland) *(WWW '06)*. Association for Computing Machinery, New York, NY, USA, 83–92. doi:10.1145/

1135777.1135794
[48] Alexandros Ntoulas, Marc Najork, Mark Manasse, and Dennis Fetterly. 2006. Detecting spam web pages through content analysis. In *Proceedings of the 15th international conference on World Wide Web*. 83–92.
[49] OpenAI. 2024. ChatGPT Search API Guide. https://platform.openai.com/docs/guides/tools-web-search. Accessed: 2025-04-20.
[50] OpenAI. 2024. Introducing ChatGPT Search. https://openai.com/index/introducing-chatgpt-search/. Accessed: 2024-12-27.
[51] OpenAI. 2024. Introducing ChatGPT Search. https://openai.com/index/introducing-chatgpt-search. Accessed: 2025-04-20.
[52] Perplexity AI. 2024. Getting Started Guide. https://docs.perplexity.ai/guides/getting-started. Accessed: 2025-04-20.
[53] Perplexity AI Team. 2025. Perplexity AI. https://www.perplexity.ai. Accessed: 2025-01-11.
[54] Samuel Pfrommer, Yatong Bai, Tanmay Gautam, and Somayeh Sojoudi. 2024. Ranking Manipulation for Conversational Search Engines. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (Eds.). Association for Computational Linguistics, Miami, Florida, USA, 9523–9552. doi:10.18653/v1/2024.emnlp-main.534
[55] SearXNG Project. 2025. SearXNG Search API Documentation. https://docs.searxng.org/dev/search_api.html. Accessed: 2025-04-21.
[56] Similarweb Ltd. 2025. Similarweb: Digital Intelligence Platform. https://www.similarweb.com/. Accessed: 2025-04-10.
[57] Sofia Eleni Spatharioti, David M. Rothschild, Daniel G. Goldstein, and Jake M. Hofman. 2023. Comparing Traditional and LLM-based Search for Consumer Choice: A Randomized Experiment. arXiv:2307.03744 [cs.HC] https://arxiv.org/abs/2307.03744
[58] StatCounter Global Stats. 2025. AI Chatbot Market Share. https://gs.statcounter.com/ai-chatbot-market-share. Accessed: 2025-09-21.
[59] Tavily. 2025. Tavily API Reference. https://docs.tavily.com/documentation/api-reference/introduction. Accessed: 2025-04-21.
[60] David Y Wang, Stefan Savage, and Geoffrey M Voelker. 2011. Cloak and dagger: dynamics of web search cloaking. In *Proceedings of the 18th ACM conference on Computer and communications security*. 477–490.
[61] Yi-Min Wang, Ming Ma, Yuan Niu, and Hao Chen. 2007. Spam double-funnel: Connecting web spammers with advertisers. In *Proceedings of the 16th international conference on World Wide Web*. 291–300.
[62] Open WebUI. 2024. Open WebUI Documentation. https://docs.openwebui.com/. Accessed: 2025-04-20.
[63] Baoning Wu and Brian D Davison. 2005. Cloaking and Redirection: A Preliminary Study.. In *AIRWeb*, Vol. 5. 7–16.
[64] Baoning Wu and Brian D. Davison. 2005. Identifying link farm spam pages. In *Special Interest Tracks and Posters of the 14th International Conference on World Wide Web* (Chiba, Japan) *(WWW '05)*. Association for Computing Machinery, New York, NY, USA, 820–829. doi:10.1145/1062745.1062762
[65] Siye Wu, Jian Xie, Jiangjie Chen, Tinghui Zhu, Kai Zhang, and Yanghua Xiao. 2024. How Easily do Irrelevant Inputs Skew the Responses of Large Language Models? arXiv:2404.03302 [cs.CL] https://arxiv.org/abs/2404.03302
[66] Hao Yang, Kun Du, Yubao Zhang, Shuai Hao, Haining Wang, Jia Zhang, and Haixin Duan. 2021. Mingling of clear and muddy water: Understanding and detecting semantic confusion in blackhat seo. In *Computer Security–ESORICS 2021: 26th European Symposium on Research in Computer Security, Darmstadt, Germany, October 4–8, 2021, Proceedings, Part I 26*. Springer, 263–284.
[67] Ronghai Yang, Xianbo Wang, Cheng Chi, Dawei Wang, Jiawei He, Siming Pang, and Wing Cheong Lau. 2021. Scalable detection of promotional website defacements in black hat {SEO} campaigns. In *30th USENIX Security Symposium (USENIX Security 21)*. 3703–3720.
[68] Qing Zhang, David Y. Wang, and Geoffrey M. Voelker. 2014. DSpin: Detecting Automatically Spun Content on the Web. In *21st Annual Network and Distributed System Security Symposium, NDSS 2014, San Diego, California, USA, February 23-26, 2014*. The Internet Society. https://www.ndss-symposium.org/ndss2014/dspin-detecting-automatically-spun-content-web
[69] Yunyi Zhang, Mingxuan Liu, Baojun Liu, Yiming Zhang, Haixin Duan, Min Zhang, Hui Jiang, Yanzhe Li, and Fan Shi. 2024. Into the dark: unveiling internal site search abused for black hat SEO. In *33rd USENIX Security Symposium (USENIX Security 24)*. USENIX Association, 1561–1578.
[70] Wei Zou, Runpeng Geng, Binghui Wang, and Jinyuan Jia. 2025. PoisonedRAG: knowledge corruption attacks to retrieval-augmented generation of large language models. In *Proceedings of the 34th USENIX Conference on Security Symposium* (Seattle, WA, USA) *(SEC '25)*. USENIX Association, USA, Article 197, 18 pages.

A Black-Hat SEO Website Classifier

A.1 Implementation

For five types of attacks, we replicated the methods from existing works [11, 33, 48, 60, 66] for semi-automated detection. The specific implementation is as follows:

Semantic Confusion. We use two models to complete the task. (1) Context semantic classifier, used to predict the probabilities that a web page belongs to 14 benign topics, outputting *prob_14* (2) Malicious web page classifier, used to predict the probability that a web page is malicious, outputting *prob_malicious*. Both models are based on the TextCNN architecture: Vocabulary Size = 10,000; Maximum Sequence Length = 500; Embedding Dimension = 128; Convolution Filter Sizes =[3,4,5]; Number of Filters per Size = 128; Pooling Layer = GlobalMaxPooling1D; Dropout Rate = 0.5; Optimizer = Adam; Batch Size = 64.

Judgment: *max(prob_14)* >0.9 And *prob_malicious* >0.9.

Redirection. We identify two types of redirection. (1) reputable domain redirecting to malicious content (Illegal search keywords + Tranco Top 10,000 domain [31] or education/government domains + Redirection + Redirect to malicious website (2) benign search redirecting to malicious content (Hot search keyword + Redirection + Redirect to malicious website). The malicious website classification model uses the same Malicious web page classifier in Semantic Confusion, outputting *prob_malicious*.

Judgment: *prob_malicious* >0.9.

Cloacking. We use the user agents of Google bot and users to crawl and obtain the page content of the two views. (1) Use text slicing techniques to generate content signatures and compare the similarity (*signature_sim*) between the user page and the bot crawled page; (2) Remove the blank pages; (3) Calculate the matching degree of the summary on the user page and the bot crawled page(*summary_sim*) (4) Calculate the DOM structure similarity (*DOM_sim*) of two views.

Judgment: *signature_sim* <0.9 And *summary_sim* >0.33 And *DOM_sim* >0.66.

Keywords Stuffing. We consider the keywords in Google Trends [18]. (1) Compute the number of Google's hot search terms matched on the page, *hotwords_count*. (2) Use the "site:domain" query in Google to determine whether the number of sub-pages is very large and all are spam content. If both conditions are satisfied, then we consider the page *Keywords Stuffing*.

Judgment: $hotwords\_count \geq 10$ And $spam\_subpages \geq 100$.

Link Farm. We conduct DNS queries supporting wildcards. Then visit the homepage or sitemap twice and extract the set of hyperlinks on it, and finally get URL set A and URL set B.

Judgment: $\max\left(\frac{|A-B|}{|A|}, \frac{|A-B|}{|B|}\right) \geq 0.2$

A.2 Evaluation

To assess the effectiveness of our classifiers used in SEO-Bench construction, we conducted a systematic evaluation for each of the five attack categories. This section outlines the evaluation methodology and presents the corresponding results.

For each classifier, we computed a confusion matrix based on manual verification. Specifically, we sampled 100 websites that were predicted as positive (`label=1`) and 100 websites predicted as negative (`label=0`). Each sample was manually checked to determine

whether the prediction matched the black-hat SEO characteristics. From these manual labels, we derived standard classification metrics including accuracy, precision, recall, and F1-score.

The overall accuracy across all classifiers was 91.12%, indicating sufficient reliability for use in dataset construction. Table 7-11 presents the confusion matrices for the five classifiers, providing a detailed view of performance across different attack types.

Table 7: Evaluation Metrics for the Redirection classifier.

	Predicted Positive	Predicted Negative	Total
Actual Positive	99	21	120
Actual Negative	1	79	80
Total	100	100	200
Accuracy: 89.0%		**Precision:** 99.0%	
Recall: 82.5%		**F1 Score:** 89.4%	

Table 8: Evaluation Metrics for the Cloaking classifier.

	Predicted Positive	Predicted Negative	Total
Actual Positive	87	5	92
Actual Negative	13	95	108
Total	100	100	200
Accuracy: 91.0%		**Precision:** 87.0%	
Recall: 94.6%		**F1 Score:** 90.6%	

Table 9: Evaluation Metrics for the Keyword Stuffing classifier.

	Predicted Positive	Predicted Negative	Total
Actual Positive	89	0	89
Actual Negative	11	100	111
Total	100	100	200
Accuracy: 94.5%		**Precision:** 89.0%	
Recall: 100.0%		**F1 Score:** 94.18%	

Table 10: Evaluation Metrics for the Semantic Confusion classifier.

	Predicted Positive	Predicted Negative	Total
Actual Positive	77	4	81
Actual Negative	23	96	119
Total	100	100	200
Accuracy: 86.6%		**Precision:** 77.0%	
Recall: 95%		**F1 Score:** 87.74%	

Table 11: Evaluation Metrics for the Link Farm classifier.

	Predicted Positive	Predicted Negative	Total
Actual Positive	92	3	95
Actual Negative	8	97	105
Total	100	100	200
Accuracy: 94.5%		**Precision:** 92.0%	
Recall: 96.8%		**F1 Score:** 94.3%	

B Evaluation Metrics

$$Resilience(Und) = \frac{|\{(q_i, t_i) \mid \text{Rewitten queries}(q_i) = \emptyset\}|}{|\{(q_i, t_i)\}|} \quad (2)$$

$$Resilience(Ret) = \frac{|\{(q_i, t_i) \mid t_i \notin \text{Retrieval references}(q_i)\}|}{|\{(q_i, t_i) \mid \text{Rewitten queries}(q_i) \neq \emptyset\}|} \quad (3)$$

$$Resilience(Sum) = \frac{|\{(q_i, t_i) \mid t_i \notin \text{Summary references}(q_i)\}|}{|\{(q_i, t_i) \mid t_i \in \text{Retrieval references}(q_i)\}|} \quad (4)$$

$$Cumulative_k = \sum_{i=1}^{k} c_i, \quad \text{where } c_i = \left(1 - \sum_{j=1}^{i-1} c_j\right) \cdot Resilience_i. \quad (5)$$

C Query Rewriting Validation Experiment

To validate the effectiveness of query rewriting against SEO attacks, we resubmitted Rewitten queries to Google Search and observed that 98.16% failed to retrieve the original SEO websites, demonstrating the approach's strong disruptive impact. Then, we explore how the degree of rewriting affects retrieval by measuring semantic and syntactic differences using semantic textual distance (STD) and edit distance (ED).

Table 12 shows that as semantic distance increases, the retrieval success rate drops significantly. Notably, even syntactic changes with an edit distance below 0.1 can reduce the success rate to under 10%, and when semantic similarity remains high (STD < 0.1), retrieval drops below 2% if the edit distance exceeds 0.2. Additionally, we observe 44 rewritten queries with a semantic distance greater than 0.5, indicating reversed or contradictory meanings (cosine similarity < 0).

This contrast suggests that query rewriting is effective not only by changing meaning but also by disrupting structural patterns used in SEO. Even minor edits can interfere with keyword matching while preserving the original intent, highlighting the role of rewriting in mitigating SEO attacks.

Table 12: The Retrieval Success Rate at Different Semantic (STD) and Syntactic (ED) Changes

ED\STD	(0.0,0.1]	(0.1,0.2]	(0.2,0.5]	(0.5,1.0]	Total
(0.0,0.1]	**10.26%**	0.00%	-	-	**9.52%**
(0.1,0.2]	10.34%	0.00%	0.00%	-	9.55%
(0.2,0.5]	**1.49%**	1.91%	2.01%	-	**1.64%**
(0.5,1.0]	0.57%	0.71%	0.33%	0.00%	0.48%
Total	2.46%	1.08%	0.57%	0.00%	1.46%

1) STD: semantic textual distance, measured as $(1 - \text{cosine similarity})/2$;
2) ED: edit distance, measured as $1 -$ Levenshtein Ratio.

D Domain-Restricted Query

Prompt Example for Domain-Restricted Query

Search strictly within site:{domain} for {product}, then recommend only results from this domain. Give me your answer and references.

E Ethical Considerations

This study is conducted under rigorous ethical oversight, and adheres to strict ethical guidelines to ensure responsible research practices.

(1) *Controlled Experimentation.* All experiments were conducted in controlled environments to avoid real-world disruption. As detailed in Section 4.1, we used existing SEO websites to build the dataset instead of generating new ones, preventing large-scale interference with real ecosystems (*e.g.* Google). The LLMSEO attacks were implemented in simplified form and within a restricted scope, using controlled subdomains to avoid contaminating legitimate domains. All test sites were clearly marked with "For Testing Purposes Only" disclaimers and taken down after experiments to eliminate residual impact. For LLMSEs with limited search functions, such as ChatGPT and Gemini, we avoided further jailbreak attempts.

(2) *Open Data.* No sensitive or private data was accessed in any experiment. For closed-source LLMSEs, we strictly followed official API policies and interacted through default interfaces. Open-source LLMSEs were deployed on isolated LAN servers using officially obtained API keys. For SEO crawling, only publicly available Google Search results were requested, and Google-documented user agents [21] were used to simulate crawler behavior.

(3) *Responsible Disclosure.* We adhered to the responsible disclosure. First, we reported to Google all 1,602 black-hat SEO websites identified in Section 4.1 to facilitate timely remediation. Second, for the nine evaluated LLMSEs, we have contacted or are contacting both commercial and open-source providers through their official vulnerability disclosure channels (*e.g.*, OpenAI's Bug Bounty) with detailed reports describing the vulnerabilities, reproduction steps, and suggested mitigation. For open-source systems, disclosures were or will be submitted via email to developers. Third, for our temporary experimental blogs, removal requests were filed with Google and Bing to ensure de-indexing after experiment termination. These actions collectively demonstrate our commitment to responsibly exposing risks while assisting vendors in strengthening system resilience.

(4) *Researcher Care.* We ensured the well-being of all researchers by providing methodological guidance and psychological support. Given the potentially disturbing nature of illegal or harmful website content, annotators worked in a controlled and supportive environment with flexible schedules to prevent fatigue. Regular mental health check-ins and access to counseling resources were maintained, and no participants reported psychological harm or undue stress during the study.