SAGEO Arena: A Realistic Environment for Evaluating Search-Augmented Generative Engine Optimization

Sunghwan Kim
Department of Artificial Intelligence
Yonsei University
Seoul, Republic of Korea
happysnail06@yonsei.ac.kr

Wooseok Jeong
Department of Computer Science and Engineering
Konkuk University
Seoul, Republic of Korea
jws010825@konkuk.ac.kr

Serin Kim
Department of Artificial Intelligence
Yonsei University
Seoul, Republic of Korea
kimserin@yonsei.ac.kr

Sangam Lee
Department of Artificial Intelligence
Yonsei University
Seoul, Republic of Korea
salee@yonsei.ac.kr

Dongha Lee*
Department of Artificial Intelligence
Yonsei University
Seoul, Republic of Korea
donalee@yonsei.ac.kr

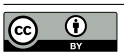
Abstract

Search-Augmented Generative Engines (SAGE) have emerged as a new paradigm for information access, bridging web-scale retrieval with generative capabilities to deliver synthesized answers. This shift has fundamentally reshaped how web content gains exposure online, giving rise to Search-Augmented Generative Engine Optimization (SAGEO), the practice of optimizing web documents to improve their visibility in AI-generated responses. Despite growing interest, no evaluation environment currently supports comprehensive investigation of SAGEO. Existing benchmarks lack end-to-end visibility evaluation of optimization strategies, operating on predetermined candidate documents and abstracting away retrieval and reranking stages in SAGE. They also discard structural information (e.g., schema markup) present in real web documents, overlooking the rich signals that search systems actively leverage in practice. Motivated by these gaps, we introduce SAGEO Arena, a realistic and reproducible environment for stage-level SAGEO analysis. SAGEO Arena integrates a full generative search pipeline over a large-scale corpus of web documents with rich structural information, enabling the first empirical analysis of how optimization signals propagate from retrieval to generation. Our findings reveal that existing approaches remain largely impractical, often degrading visibility in retrieval and reranking. Leveraging structural information helps mitigate these limitations, yet effective SAGEO requires tailoring optimization to each pipeline stage. Overall, our benchmark provides a practical foundation for advancing SAGEO.

CCS Concepts

• **Information systems** → **Web searching and information discovery**; **Page and site ranking**; **Evaluation of retrieval results**; *Retrieval models and ranking*; *Language models.*

*Corresponding author



KDD 2026, Jeju Island, Republic of Korea.

ACM ISBN 979-8-4007-2259-2/2026/08
https://doi.org/10.1145/3770855.3818146

Keywords

Search-Augmented Generative Engine; Generative Engine Optimization; Search Engine Optimization; Benchmark; Evaluation

ACM Reference Format:
Sunghwan Kim, Wooseok Jeong, Serin Kim, Sangam Lee, and Dongha Lee. 2026. SAGEO Arena: A Realistic Environment for Evaluating Search-Augmented Generative Engine Optimization. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD 2026), August 9–13, 2026, Jeju Island, Republic of Korea.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3770855.3818146

Resource Availability:
The source code and artifact of this paper have been made publicly available at https://doi.org/10.5281/zenodo.20423531, with the GitHub repository at https://github.com/happysnail06/SAGEO_Arena.

1 Introduction

With the rapid advancement of Large Language Models (LLMs), Search-Augmented Generative Engines (SAGE) [4, 22, 23, 49] have emerged as a new paradigm for information access. By integrating large-scale retrieval with generative capabilities, these systems perform *generative search*, providing synthesized answers that directly address user queries. As an increasing portion of web traffic now originates from such AI-generated answers [8], how content gains exposure online is fundamentally changing. This shift has given rise to Search-Augmented Generative Engine Optimization (SAGEO), the practice of optimizing web documents to improve their visibility (i.e., presence and contribution) in AI-generated responses.

SAGEO naturally extends the core objective of traditional Search Engine Optimization (SEO) [18]. For decades, optimization has focused on improving retrieval and ranking on Search Engine Results Pages (SERPs), primarily through on-page factors such as titles and metadata [43]. With the emergence of SAGE, optimization has increasingly shifted toward a generation-centric perspective, studying how a document is favored and cited in model responses [7]. A prominent line of study is Generative Engine Optimization (GEO) [1], focusing on presentational modifications to improve visibility at the generation stage. Despite this shift, generative search often builds on traditional search systems. For example,

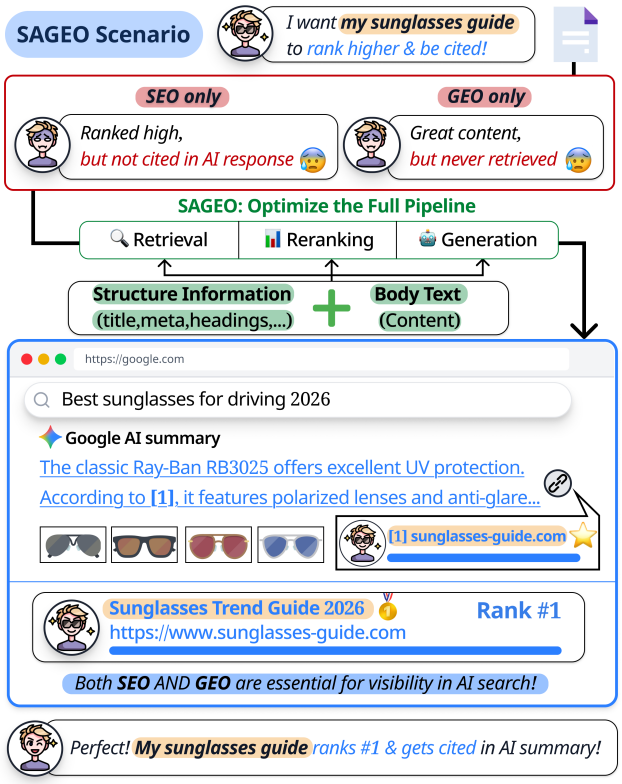


Figure 1: Optimizing for ranking alone (SEO) or citation alone (GEO) causes failure at either retrieval/reranking or generation. SAGEO jointly optimizes structural information and body text across the full pipeline to succeed at both stages.

MS Copilot relies on Bing Search to ground its responses [35], meaning that documents must first be surfaced by the underlying search pipeline before they can enter the LLM's context. Thus, SEO and GEO must be jointly addressed to achieve successful optimization. In light of this, we clearly distinguish SAGEO as optimization that targets the full generative search pipeline, from retrieval and reranking to generation (Figure 1). This motivates a systematic study of what constitutes effective SAGEO and how optimization signals propagate and interact across stages in generative search.

Despite growing interest, there exists no evaluation environment that enables such investigation. Existing benchmarks [1, 9, 38, 50] tend to oversimplify real web environments, leaving it unclear whether current optimization strategies are effective in practical settings. Specifically, there are two major limitations: **(1) Lack of end-to-end evaluation.** Current benchmarks predominantly operate on pre-determined candidate documents, abstracting away the retrieval and reranking stages that precede generation. It remains unclear whether current optimization strategies trade off or benefit performance at earlier search stages, making stage-level analysis of SAGEO essential. **(2) Loss of structural information.** Existing benchmarks operate solely on plain body text, discarding structural information present in real web documents (e.g., schema markup). In practice, real search systems do not determine visibility from body text alone. Structural information continues to play an important role in search-stage visibility, while similar structured representations, such as markdown summaries, have also been shown to improve generation-stage visibility [38]. Yet no existing benchmark preserves structural information, leaving its impact across the generative search pipeline largely unexplored.

Motivated by these gaps, we introduce SAGEO Arena, a realistic and reproducible benchmark designed for stage-level SAGEO analysis. Crucially, our benchmark is distinguished in two ways: **(1) Comprehensive generative search environment.** Unlike existing benchmarks that assume fixed retrieval context, SAGEO Arena integrates a full generative search pipeline where documents are dynamically retrieved, ranked, and passed to generation. Each stage is modular and configurable, allowing researchers to trace how optimization signals propagate across the complete search process. **(2) Extensive corpus with rich structural annotations.** We curate a corpus of 170k web documents spanning 9 domains. Following guidelines published by commercial search engines, we extract both body text and rich structural information that SAGEO must navigate in practice (e.g., meta descriptions, headings, and schema markup). Overall, we aim to construct an environment that closely reflects real-world deployment, paving the way for optimization strategies that generalize beyond controlled settings.

We conduct extensive experiments on SAGEO Arena to evaluate how optimization strategies affect document visibility across the full generative search pipeline. Our analysis reveals that body text optimization alone, the predominant focus of prior work, remains largely insufficient in realistic generative search settings. It provides only marginal gains in generation-stage visibility and, more importantly, degrades retrieval performance, causing optimized documents to drop out of the retrieval results. By contrast, structural information helps mitigate this degradation by preserving signals that support document visibility in early pipeline stages. We also find that the two scopes play fundamentally complementary roles, with structural information driving retrieval and informative body text remaining a key factor for reranking and generation. Through in-depth analysis across domains, citation behavior, and optimization models, we further show that each pipeline stage prioritizes different document qualities, motivating the need for stage-aware optimization. Guided by these findings, we propose stage-aware SAGEO, a method that tailors optimization strategies to each stage, achieving the strongest overall visibility gain among all evaluated strategies. To summarize, our contributions are as follows:

- We introduce SAGEO Arena, the first benchmark that enables stage-level visibility evaluation of SAGEO. By integrating a full generative search pipeline with a large-scale corpus preserving rich structural information, SAGEO Arena bridges the gap between existing benchmarks and real-world SAGE.
- We provide the first empirical analysis of structural information optimization, studying how structural signals that generative search systems must navigate in practice influence visibility.
- Through extensive experiments, we show that current optimization approaches are insufficient in realistic settings. To address this, we introduce stage-aware SAGEO, providing actionable guidance to further facilitate SAGEO research.

Table 1: A comparison of SAGEO Arena to existing benchmarks.

Benchmark	Environment				Document Details		Evaluation		
	Doc. Corpus	Retrieval	Reranking	Generation	Structure Info.	Body Text	Search	Generation	Visibility Metric
GEO-Bench [1]	-	✗	✗	✓	✗	✓	✗	✓	Word Count
AutoGEO [50]	-	✗	✗	✓	✗	✓	✗	✓	Word Count, Utility
C-SEO Bench [38]	-	✗	✗	✓	✗	✓	✗	✓	Citation Rank
CC-GSEO-Bench [9]	-	✗	✗	✓	✗	✓	✗	✓	Influence
SAGEO Arena (Ours)	170K	✓	✓	✓	✓	✓	✓	✓	Hit Rate, Rank Change

2 Related Work

2.1 Generative Search Engine Optimization

With the rise of LLMs, generative search [6, 20, 31, 56] has become a cornerstone of modern information access, offering synthesized answers grounded in retrieved documents. This shift has introduced new visibility challenges for content creators, motivating research on optimization strategies tailored to generative engines. Existing work can be broadly categorized into two directions: (1) *Black-hat* approaches explore adversarial methods that exploit model behavior through malicious content injection [24, 36, 37]. In contrast, (2) *white-hat* approaches focus on cooperative content modification, formalized as GEO [1]. Building on this foundation, subsequent work [9, 38, 50] has advanced GEO research by proposing diverse optimization strategies and evaluation dimensions (Table 1). Nevertheless, existing evaluation protocols often rely on oversimplified environments, predominantly assessing optimization effects at the generation stage with fixed document candidates. In other words, they assume that the target document remains within the LLM context after optimization, without testing whether the optimized document can still be retrieved and reranked into that context. Thus, how optimization signals propagate through the full generative pipeline remains largely unexplored. Motivated by these, this study provides a comprehensive evaluation environment that explicitly models the end-to-end generative search pipeline.

2.2 Comparison of SEO and GEO

Traditional SEO has long served as the dominant framework for online visibility [17, 26, 39, 43], aiming to improve a source's position on SERP. Following established guidelines from leading search engine providers [14, 34], SEO has long emphasized structural information that helps pages be interpreted and surfaced, including titles, meta descriptions, headings, and schema markup. In contrast, GEO research predominantly focuses on modifying body text in the generation stage [10], investigating rewriting strategies that adjust textual properties such as fluency, technicality, and ease of understanding. However, recent empirical findings [38] challenge this generation-centric perspective, demonstrating that a document's position determined in the retrieval and reranking stage plays a far more dominant role in determining its visibility in the final response. Notably, document position is often shaped by structural information emphasized in traditional SEO, which raises a fundamental question: *do these signals also contribute to visibility in generative engines?* Existing benchmarks offer limited insight into this question, and we bridge this gap by jointly studying how SEO and GEO signals interact across the full generative search pipeline.

2.3 Structured Web Information Understanding

The World Wide Web is fundamentally built upon HyperText Markup Language (HTML). Beyond body content, HTML documents encode structural information through elements such as title tags, meta descriptions, heading hierarchies, and schema markup. These structural attributes have long influenced how search engines crawl, index, and rank documents [28]. Recent studies [11, 19, 46, 48] demonstrate that structural information helps LLMs better understand web content, showing that structural representations in HTML complement plain text in Retrieval-Augmented Generation (RAG) systems. Moreover, there is evidence indicating that commercial SAGEs, such as MS Copilot and Google Search, similarly leverage structural information when interpreting and surfacing web documents [15, 25, 34]. These findings motivate incorporating structural information into SAGEO research, but no existing benchmark systematically evaluates its impact. We address this gap by proposing a novel benchmark that preserves the structural elements in web documents, enabling more realistic evaluation of optimization strategies in real-world generative search settings.

3 SAGEO Arena

Problem Setting. We consider a search-augmented generative engine that answers user queries by retrieving relevant web documents and synthesizing a response grounded in the retrieved content. The pipeline follows the well-established RAG paradigm [29], consisting of a retriever, reranker, and generator. In this setting, a document must first be retrieved and ranked before it can influence the generated response. This creates an inherent dependency between traditional SEO concerns (retrieval and ranking) and emerging GEO concerns (generation). Our benchmark is designed to capture this full pipeline, enabling thorough evaluation of optimization strategies across all three stages.

Design Principles. Our core challenge lies in reproducing an environment that reflects real-world search-augmented generative engines. To achieve this, we (1) adopt a standardized search-augmented generation pipeline following established practices [13, 16, 40] and (2) preserve structural information explicitly recommended by commercial search engines for optimization [14, 33, 34].

Task Formulation. Let q denote a user query and $\mathcal{D}$ a document corpus. Given q, the system produces a response A_q through three stages: (1) a retriever $\mathcal{R}$ returns a candidate set $\mathcal{D}_q = \mathcal{R}_k(q, \mathcal{D})$ containing the top-k documents; (2) a reranker $\mathcal{F}$ reorders $\mathcal{D}_q$ by relevance, producing a ranked list $\mathcal{D}_q^* = \mathcal{F}(q, \mathcal{D}_q)$; (3) a generator

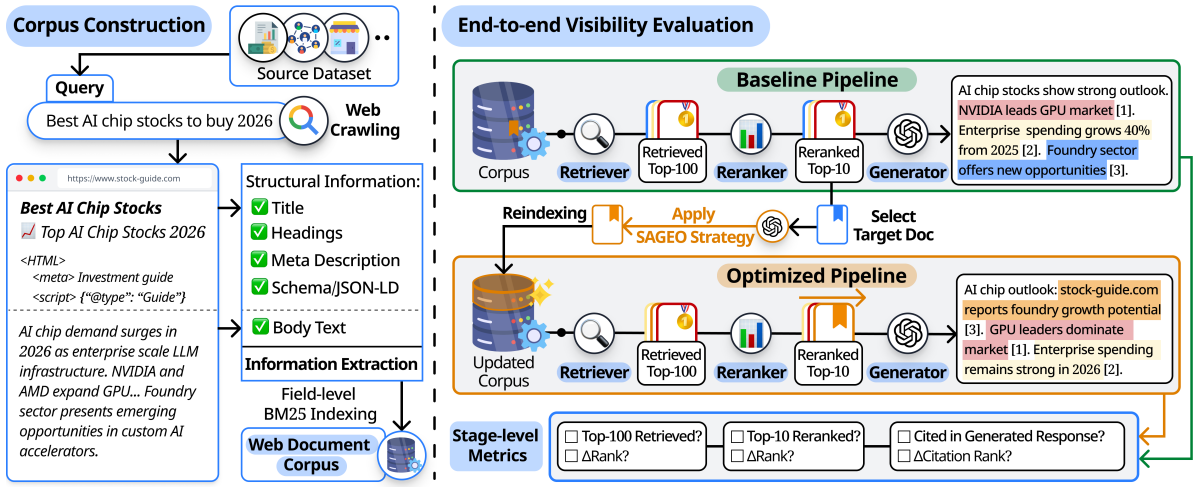


Figure 2: Overview of SAGEO Arena. During document corpus construction, we extract structural information (title, meta description, headings, schema) and body text from web documents. During SAGEO evaluation, we first execute the pipeline with each test query and randomly select a target document from those that reach the generation stage (top-k after reranking) as the baseline. The target document is then optimized, re-indexed into the corpus, and the pipeline is re-executed with the same query, tracking whether the document maintains, gains, or loses visibility at each stage (retrieval, reranking, generation).

$\mathcal{G}$ produces a response $A_q = \mathcal{G}(q, \mathcal{D}_q^*)$ with inline citations.

$$A_q = \mathcal{G}\Big(q,\ \mathcal{F}\big(q,\ \mathcal{R}_k(q, \mathcal{D})\big)\Big). \tag{1}$$

The visibility of each document $d_i \in \mathcal{D}_q^*$ is reflected by whether and to what extent it is cited in A_q. Let $d^{\text{tgt}} \in \mathcal{D}$ denote a target document relevant to the query q. SAGEO aims to optimize d^{tgt}, without prior knowledge of the query q, such that it (1) is retrieved into $\mathcal{D}_q$, (2) ranks highly in $\mathcal{D}_q^*$, and (3) maximizes visibility in A_q.

3.1 Corpus Construction

To create a large-scale corpus with strong domain diversity, we curate queries from nine established information retrieval datasets spanning a broad range of domains (Table 6). We sample 300 queries from each dataset, yielding 2,700 unique queries in total. Following prior approaches, we retrieve up to 100 search results per query using the Google Custom Search API. We then crawl each URL to extract both body text and rich structural information (Section 3.2). After filtering documents with malformed content, the final corpus contains 171,003 unique web documents. On average, each query is associated with 63 candidate documents. The 300 sampled queries per domain serve as test queries for evaluating optimization. Detailed statistics of SAGEO Arena are provided in Appendix A.

3.2 Structural Information Extraction

Unlike prior benchmarks that extract only body text, we explicitly preserve structural information embedded in real-world web documents. However, such documents exhibit substantial diversity and noise [47, 54], making it impractical to preserve all structural information for systematic evaluation. We therefore selectively retain structural fields consistently emphasized in public search engine guidelines [34], enabling a realistic yet controlled study of how these elements influence search-augmented generative engines.

- **Title.** The document title provides a concise topical summary of the page. It serves as a primary signal of the page's main topic and intent, playing a key role in initial relevance assessment.
- **Meta Description.** A concise page summary intended for search result snippets. This field offers a compact yet informative description of page content that complements the title.
- **Headings.** Hierarchical section headers (H1–H6) that signal topical structure within a document. Search engines treat headings as strong indicators of what each section is about, often weighting them higher than surrounding body text.
- **Schema/JSON-LD.** Structured data markup that explicitly defines entities, attributes, and relationships in a machine-readable format. Search engines rely on this markup to interpret page semantics that cannot be inferred from plain text alone.
- **Body Text.** The main textual content of the document, representing the primary source of information.

Notably, merging these fields into a unified text representation at the indexing stage would obscure the distinct signals carried by each component. Therefore, search-augmented generative engines broadly index these fields as separate components and combine them later during relevance scoring [42]. We adopt the same practice, preserving field-level separation to enable fine-grained analysis of how each component contributes to visibility across the pipeline.

3.3 Search-Augmented Generative Engine

We implement a full generative search engine pipeline with retrieval, reranking, and generation. Each stage reflects real-world practices while remaining modular for controlled experimentation.

Document Representation. In our setting, a web document consists of two types of content: structural information and body text. The structural information of a document, denoted as $\mathcal{S}(d) = \{s_1, \ldots, s_m\}$, is a set of structural fields corresponding to the elements described in Section 3.2 (e.g., title, meta description). The

body text of a document, denoted as $\mathcal{P}(d) = \{p_1, \ldots, p_n\}$, is a set of passages chunked from the body text following standard practice. A document can therefore be represented as $(\mathcal{S}(d), \mathcal{P}(d))$.

Semantic Unit. In generative search systems, documents are often chunked and retrieved at the passage level. However, structural information of a web document is shared across all passages, making it essential to bridge this gap between document-level and passage-level information for effective retrieval. We therefore pair each passage $p_i \in \mathcal{P}(d)$ with the associated structural information $\mathcal{S}(d)$, creating a semantic unit $\mathcal{U}(p_i) = \mathcal{S}(d) \cup \{p_i\}$ that serves as a passage representation of p_i during retrieval and reranking.

Retriever. We employ BM25 [41], a widely adopted lexical retrieval method, as our retriever $\mathcal{R}$. This choice reflects the practical role of lexical retrieval as an efficient and scalable first-stage candidate generator in web-scale search systems. For each semantic unit $\mathcal{U}(p)$, we build a separate BM25 index for every element $u \in \mathcal{U}(p)$. Given a query q, we score q against each u independently, and aggregate the resulting ranks via reciprocal rank fusion to obtain the retrieval score of passage p:

$$\text{score}_{\mathcal{R}}(q, p) = \sum_{u \in \mathcal{U}(p)} \frac{1}{\kappa + \text{rank}(q, u)}, \tag{2}$$

where $\text{rank}(q, u)$ is the rank of element u, and κ is a constant. The top-k passages are selected as candidates for reranking.

Reranker. The retrieved candidates are reranked using a cross-encoder that scores each query-passage pair (q, p) independently. Following practices in production systems (e.g., Vespa AI[1], Google Vertex AI[2]), we score relevance at the passage level rather than over entire documents. This approach handles cases where only specific sections of a long document are relevant to the query.

Generator. The top reranked candidates are provided to the generation model along with their structural fields. The model is prompted to cite sources, enabling measurement of document-level visibility.

By grounding our design in a complete pipeline with structurally enriched documents, SAGEO Arena enables analyses beyond the scope of existing benchmarks. Specifically, it allows (1) stage-wise evaluation of diverse strategies, (2) component-level ablations, and (3) testing robustness and generalization across realistic settings.

3.4 Stage-level Visibility Evaluation

We describe how SAGEO Arena measures document visibility across each pipeline stage and quantifies the effect of optimization.

Evaluation Pipeline. As shown in Figure 2, we establish a baseline as follows. For each test query q, we first execute the full generative search pipeline in SAGEO Arena. Among the documents that reach the generation stage (i.e., ranked within the top-k at the reranking stage), we randomly select one as the target document d^{tgt}, yielding the baseline instance (q, d^{tgt}). The target document d^{tgt} is then optimized using a specified strategy and reindexed into the corpus following the process described in Section 3.3. We then re-execute the search pipeline with the same query and compare the document's visibility against the baseline at each stage. This enables evaluating optimization effects under realistic, stage-level settings, rather than within a fixed retrieval context like prior benchmarks.

Tracking Target Document. Since search-augmented generative engines internally operate at the passage level, multiple passages from the same document may appear in the candidate list as they are scored independently. This makes tracking the visibility of a target document an inherent challenge. To address this, we define the rank of a target document d_q^{tgt} as the highest rank achieved by any of its associated passages at each pipeline stage:

$$\text{rank}(d_q^{\text{tgt}}) = \min_{p \in \mathcal{P}(d_q^{\text{tgt}})} \text{rank}(p), \tag{3}$$

where $\mathcal{P}(d_q^{\text{tgt}})$ denotes the set of passages derived from d_q^{tgt}. The passage with the highest rank serves as the visibility indicator of the associated document at each stage, and we measure optimization effectiveness in two ways: (1) whether the target document appears within the top-k candidates at each stage, and (2) how the rank of the target document shifts before and after optimization.

Metrics. For each test query $q \in Q$, we execute the pipeline and track the rank of its associated target document d_q^{tgt} via Equation (3). We then evaluate visibility at each stage using two complementary metrics: (1) Hit Rate (H@k), measuring the proportion of queries for which the target document appears within the top-k candidates. For retrieval and reranking, it is defined by

$$\text{H@}k = \frac{1}{|Q|} \sum_{q \in Q} \mathbb{I}\left(\text{rank}(d_q^{\text{tgt}}) \le k\right), \tag{4}$$

where $\mathbb{I}(\cdot)$ returns 1 if the condition is satisfied and 0 otherwise. For generation, we instead use Citation Rate, the proportion of queries where the target document is cited in the final response. (2) Rank Change, the average positional shift of the target document between baseline and optimized settings:

$$\Delta\text{Rank} = \frac{1}{|Q|} \sum_{q \in Q} \left(\text{rank}_{\text{base}}(d_q^{\text{tgt}}) - \text{rank}_{\text{SAGEO}}(d_q^{\text{tgt}})\right), \tag{5}$$

where $\text{rank}_{\text{base}}$ and $\text{rank}_{\text{SAGEO}}$ represent the target document rank before and after optimization, respectively. Documents not appearing in the top-k candidates either at the retrieval or reranking stage are assigned a default rank of $k + 1$. We set $k = 100$ for retrieval and $k = 10$ for reranking. For generation, rank at this stage is defined as the citation order in which a source first appears in the response.

4 Experimental Setup

Optimization Strategies. We evaluate 10 LLM-based optimization strategies derived from prior research. Eight are adapted from [1], each targeting a specific stylistic or content modification to improve document visibility. These include strategies that adjust tone and readability (Authoritative, Fluency, Easy Language), add supporting evidence (Cite Sources, Quotation, Statistics), and diversify vocabulary (Technical Terms, Unique Words). We additionally include All-in-One [38], which applies all eight strategies simultaneously along with structural formatting such as bolding and improved layout, and AutoGEO [50], which leverages preference rules learned from generative engine behavior to guide document optimization.

- **Authoritative:** Modifies text style to be more persuasive, making statements definitive while maintaining factual accuracy.

[1] https://vespa.ai/
[2] https://cloud.google.com/vertex-ai

Table 2: Effectiveness of SAGEO strategies. We report Hit Rate and ΔRank for each optimization target at each stage. Percentage values indicate relative change from the pre-optimization baseline. Positive ΔRank indicates rank improvement. H@10 at reranking is 1.00 in the baseline, as all target documents are selected from the top-10 candidates at the reranking stage.

Strategy	Body Text only						Structural Information only						Both					
	Retrieval		Reranking		Generation		Retrieval		Reranking		Generation		Retrieval		Reranking		Generation	
	H@20	ΔRank	H@10	ΔRank	Cite	ΔRank	H@20	ΔRank	H@10	ΔRank	Cite	ΔRank	H@20	ΔRank	H@10	ΔRank	Cite	ΔRank
Baseline	0.58 - - %	-	1.00 - - %	-	0.50 - - %	-	0.58 - - %	-	1.00 - - %	-	0.50 - - %	-	0.58 - - %	-	1.00 - - %	-	0.50 - - %	-
Auth.	**0.57** -1%	**-0.20**	**0.91** -9%	-0.24	0.49 -3%	-0.10	0.68 +18%	+1.60	0.81 -19%	-0.59	0.49 -3%	+0.01	0.69 +19%	+0.47	0.77 -23%	-0.80	0.48 -3%	+0.01
Cite	0.56 -3%	-1.50	0.87 -13%	-0.45	0.48 -4%	-0.13	0.74 +28%	+6.05	0.86 -14%	-0.17	0.52 +2%	+0.25	0.66 +13%	-1.81	0.73 -26%	-0.99	0.47 -6%	-0.04
EasyLang	0.52 -10%	-4.18	0.84 -16%	-0.54	0.49 -4%	-0.09	0.74 +27%	+5.15	0.86 -14%	**+0.13**	0.53 +5%	+0.38	0.71 +22%	+3.76	0.80 -20%	-0.30	0.51 +2%	+0.34
Fluency	**0.57** -1%	-0.71	**0.91** -9%	**-0.18**	**0.50** -1%	**-0.01**	**0.75** +30%	**+6.62**	**0.88** -12%	+0.08	0.53 +5%	+0.37	0.72 +24%	+2.37	0.81 -19%	-0.42	0.49 -2%	+0.11
Quote	**0.57** -1%	-0.33	0.90 -10%	-0.36	0.47 -7%	-0.26	0.74 +27%	+5.47	0.85 -15%	-0.28	0.53 +4%	+0.30	**0.79** +35%	**+8.87**	**0.85** -15%	**-0.29**	0.51 +2%	+0.24
Stats	**0.57** -2%	-0.51	0.90 -10%	-0.33	0.48 -4%	-0.18	**0.75** +29%	+6.03	0.86 -14%	-0.15	**0.54** +7%	**+0.47**	0.71 +22%	+1.53	0.80 -20%	-0.75	0.48 -5%	-0.08
Tech.	0.50 -14%	-6.23	0.80 -20%	-1.03	0.47 -6%	-0.15	0.66 +13%	-0.59	0.79 -21%	-0.61	0.51 +1%	+0.22	0.62 +6%	-2.39	0.64 -36%	-2.02	0.43 -14%	-0.44
Unique	0.53 -8%	-3.47	0.86 -14%	-0.70	0.46 -8%	-0.33	0.69 +19%	+1.09	0.81 -19%	-0.62	0.49 -4%	-0.06	0.66 +13%	-0.16	0.75 -25%	-1.24	0.43 -13%	-0.48
All-in-One	0.50 -14%	-5.93	0.83 -17%	-0.68	0.49 -2%	-0.03	0.71 +22%	+2.44	0.82 -18%	-0.42	0.50 -1%	+0.12	0.69 +17%	+1.96	0.77 -23%	-0.62	**0.52** +3%	**+0.38**
AutoGEO	0.37 -36%	-22.35	0.58 -42%	-2.28	0.39 -22%	-0.78	0.60 +4%	-6.68	0.73 -27%	-0.72	0.51 +0%	+0.31	0.44 -25%	-20.15	0.58 -42%	-1.93	0.44 -12%	-0.13
Avg.	0.53 -9%	-4.54	0.84 -16%	-0.68	0.47 -6%	-0.21	0.71 +22%	+2.72	0.83 -17%	-0.34	0.52 +2%	+0.24	0.67 +15%	-0.56	0.75 -25%	-0.94	0.48 -5%	-0.01

- **Cite Sources:** Adds inline citations to external sources, inserting brief references to enhance reliability and trustworthiness.
- **Fluency:** Improves grammatical correctness by refining sentence structure and transitions without altering content meaning.
- **Quotation:** Incorporates quotations from authoritative figures or sources with proper attribution to provide external validation.
- **Easy Language:** Simplifies by replacing complex vocabulary with accessible alternatives while preserving information.
- **Statistics:** Adds quantitative data and numerical facts inline within sentences to make claims more concrete and credible.
- **Technical Terms:** Introduces domain-specific terminology to present content in a more expert and authoritative manner.
- **Unique Words:** Enriches vocabulary by incorporating less common words to signal higher content quality and specialization.
- **All-in-One:** Combines all eight base strategies, then enhances text structure by bolding key features and improving layout to increase information accessibility and visual attractiveness.
- **AutoGEO:** Applies optimization rules learned from analyzing generative engine citation patterns to guide document rewriting.

Pipeline Configuration. We use BM25 as the retriever, Qwen3-Reranker-4B [55] as the reranker, and GPT-5-mini as the generator in the default SAGEO Arena pipeline. Further implementation details of the pipeline and optimization are described in Appendix B.

5 Results & Discussion

5.1 Main Results

We present findings from extensive experiments on SAGEO Arena. Our goal is to evaluate how optimization strategies affect document visibility across the full generative search pipeline. Existing research focuses on body text alone in SAGE, leaving the effectiveness of such a setting in realistic environments largely unexplored. Moreover, optimizing structural information (Section 3.2), the process of organizing core document information into metadata fields so that search systems can correctly interpret the document, has not been studied in the context of generative search. We therefore separate optimization scope into three settings: body text only, structural information only, and both, enabling the first analysis of where gains and losses originate at each generative pipeline stage.

Limitation of Body Text only Optimization. As shown in Table 2 (Left), optimizing body text alone consistently degrades visibility across all stages. At retrieval, optimization strategies that replace common expressions with domain-specific terms (e.g., technical words) or uncommon vocabulary (e.g., unique words) show the largest retrieval drops. We attribute this to the lexical mismatch between optimized documents and user queries, which typically use common vocabulary. For example, replacing terms like "*eating*" with "*alimentary routines*" or "*sleeping*" with "*somnolence*" directly reduces term overlap, causing BM25-based retrievers to assign lower relevance scores. Notably, AutoGEO [50] exhibits the largest degradation, with a retrieval rank drop of −22.35. We find that AutoGEO tends to substantially expand the document content, introducing lengthy rewrites that dilute keyword density and shift the document further from the original query vocabulary. At reranking, a similar but milder degradation is observed, suggesting that rewrites may introduce slight semantic shifts that rerankers are sensitive to. However, even when the average rank drop is modest, small shifts at retrieval or reranking can be critical in practice. In generative search, only top-ranked documents are passed to the generator, and a document falling below this threshold is excluded entirely, making it invisible at the generation stage regardless of its content quality. At generation, gains remain relatively marginal across strategies, aligning with recent findings that presentational modifications offer limited visibility improvements [38]. Overall, body-only optimization is insufficient for enhancing visibility throughout the pipeline. These findings suggest that effective optimization must extend beyond body text to structural information, which plays an important role in document ranking in real-world generative search systems.

Effectiveness of Structural Information Optimization. As shown in Table 2 (Center), extending optimization scope to structural information significantly improves visibility across all strategies compared to body-only optimization. The improvement is

Figure 3: Reranking case studies illustrating two factors that improve document ranking after optimization. Case A shows that adding content that directly addresses the query's informational need boosts rank. Case B shows that placing the direct answer in early paragraphs improves ranking position.

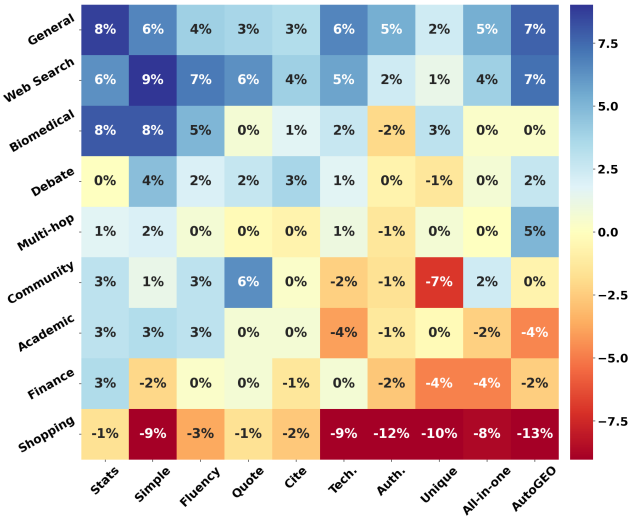


Figure 4: Effectiveness of each optimization strategy across nine domains, measured by change in citation rate at the generation stage. Positive % indicate increased citation rate.

particularly notable at retrieval, with a +22% boost in Hit Rate and a +2.72 average retrieval rank gain. Structural information is inherently designed to be dense with query-relevant terms, increasing lexical overlap with user queries that BM25-based retrievers directly prioritize. Accordingly, strategies that enrich content with keywords, entities, and numbers show strong effectiveness. For example, adding statistics rewrites a verbose meta description into a concise summary with specific facts like "*1983, comedy, Rotten Tomatoes, grossed 61M*," while adding quotations refines a generic title "*Panel Clarifies Advice*" into "*IOM Panel Clarifies Vitamin D Guidance*," introducing entity-specific terms that better align with user queries. Interestingly, optimizing structural information alone also improves visibility at reranking and generation. We attribute this partly to the increased number of documents that pass through retrieval into downstream stages, raising their chances of being reranked favorably and cited. Additionally, optimizing structural information effectively places well-organized, informative summaries at the top of the document, consistent with recent findings that such positioning benefits document visibility in the generation stage [38]. However, applying optimization to both scopes (Table 2 Right) yields lower visibility gains. The negative retrieval effects of body-text modification, observed in the body-only setting, could partially diminish the gains from optimizing structural information.

Reranking as a Persistent Bottleneck. All optimization strategies struggle at the reranking stage, showing consistent visibility degradation regardless of optimization scope (Table 2). One contributing factor is that top-ranked documents are very close in relevance, making them more sensitive to rank displacement from even minor content changes. Although the average rank drop remains within 1 position across most settings, even such small shifts can alter the relative ordering among closely ranked candidates. Notably, 5.8% of target documents in our experiments dropped from rank 10 to 11 during reranking, narrowly missing the input threshold for the generator in SAGEO Arena. This highlights the importance of maintaining or improving rank position within generative search pipelines, as they inherently impose such cutoffs. To better understand what factors positively or negatively affect reranking results, we conduct a case study analyzing documents that gained or lost rank after optimization. Figure 3 presents two representative cases. It is worth noting that SAGEO optimizes documents without access to the incoming query, meaning content modifications cannot be tailored to specific query intents. Despite this constraint, clear patterns emerge in how the reranker responds to different types of optimizations. First, the reranker favors content additions that enhance alignment with the query's informational need, while penalizing additions that expand the document's scope beyond what the query seeks. Second, placing the answer early in the document yields higher reranking scores, whereas restructuring that displaces the answer to later paragraphs results in significant rank drops, even when the answer itself remains intact. These observations indicate that reranking-aware optimization should preserve topical alignment and answer prominence over broad content expansion.

5.2 In-depth Analysis

To further understand the effectiveness of SAGEO, we analyze three dimensions: domain, citation source, and SAGEO backbone model.

Domain Analysis. Figure 4 shows the effectiveness of each optimization method across nine domains, reporting change in citation rate at the generation stage. We observe that domains with general information-seeking queries (e.g., Web Search, General QA) benefit the most, as strategies like adding statistics provide concrete

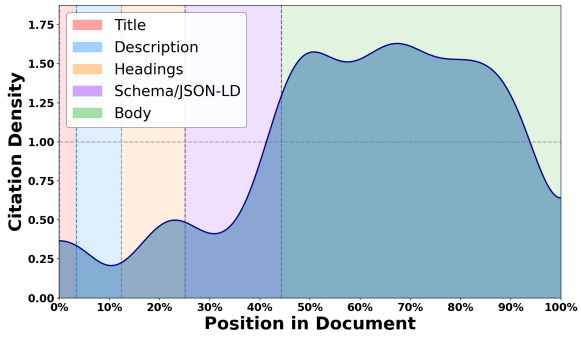


Figure 5: Density of citation sources across document regions.

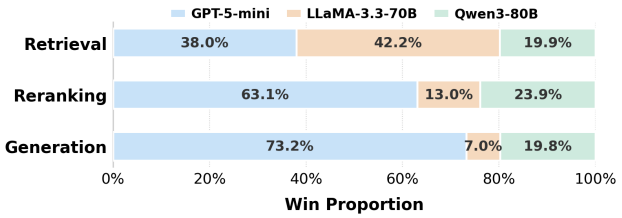


Figure 6: Win rate comparison across three backbone models (GPT-5-mini, LLaMA-3.3-70B, Qwen3-80B) at each stage. A model is considered a win when its optimized document achieves the highest rank among the three at a given stage.

Table 3: Comparison of SAGEO strategy combination and StageAware optimization (Ours) under the Both setting. StageAware achieves the strongest SAGEO performance.

Method	Retrieval		Reranking		Generation	
	H@20	ΔRank	H@10	ΔRank	Cite	ΔRank
Baseline	0.58	-	1.00	-	0.50	-
Combined Strategy						
EasyLang + Quote	0.67 +14%	-0.25	0.78	-0.33	0.54	+0.64
EasyLang + Stats	0.65 +11%	-1.77	0.76	-0.56	0.51	+0.38
Fluency + Quote	0.66 +13%	-0.63	0.77	-0.43	0.54	+0.65
Fluency + Stats	0.65 +11%	-1.63	0.76	-0.55	0.53	+0.60
StageAware (Ours)	**0.75 +28%**	**+4.86**	**0.80**	**-0.08**	**0.58**	**+1.01**

evidence that directly address these queries. Shopping is the only domain where every optimization method decreases citation likelihood. Product documents are often already well-organized for their customers, and the queries in this domain are predominantly casual and recommendation-seeking (e.g., "*gift ideas for my friend*"). Optimizing such documents may shift them away from this expected tone, reducing their competitiveness in the generator's response. These results suggest that optimization strategies applied without considering domain-specific user intent can be ineffective or even harmful, emphasizing the need for domain-aware optimization.

Citation Source Analysis. To understand what drives citation at the generation stage, we examine how generative models interact with document content when forming responses. To achieve this, we conduct an experiment prompting the generator to provide the exact quote used for each citation in its response. We then apply fuzzy matching to locate each quoted segment in the source document and identify the region from which it originates. Figure 5 illustrates the results as a density plot. Each colored region corresponds to a structural component (e.g., title, meta description, headings, JSON-LD) or body text, and values above the horizontal baseline ($y = 1.0$) indicate that the region tends to be cited more frequently. We observe that the vast majority of citations originate from body text, while structural information is cited less frequently despite its strong contribution to retrieval. We attribute this to the information density of body text, which provides richer evidence for addressing user queries compared to the concise, keyword-oriented nature of structural information. Combined with the earlier finding that structural information improves visibility at the generation stage (Section 5.1), these results suggest that structural information plays an important role in surfacing documents, but the generator primarily references body text when forming responses. Thus, structural information and body text serve complementary roles in SAGE, and effective optimization requires addressing both.

SAGEO Backbone Model Analysis. To examine whether the choice of backbone model for SAGEO exhibits distinct optimization behaviors, we conduct experiments with two additional open-source models, LLaMA-3.3-70B and Qwen3-80B (Figure 6). We compare them against GPT-5-mini, applying the All-in-One strategy across all three models. Interestingly, LLaMA-3.3-70B achieves the highest win rate (42.2%) at retrieval, surpassing GPT-5-mini (38.0%). We find that LLaMA often generates short, keyword-dense text (296 words on average vs. 782 for GPT). Because BM25 rewards keyword matches and applies length normalization that favors shorter documents, this style is naturally advantaged at the retrieval stage. However, this advantage reverses at reranking, where GPT-5-mini dominates with a 63.1% win rate while LLaMA-3.3-70B drops to 13.0%. Unlike BM25, the LLM-based reranker evaluates both relevance and coherence of the documents to the user query, and tends to penalize irrelevant content [21, 45]. A case study reveals that LLaMA-optimized documents frequently mention query terms without meaningfully addressing the underlying question. For instance, given the query "*What are some of the most useful vim shortcuts?*", LLaMA opens with an unrelated introduction while GPT directly lists shortcuts with clear formatting. At generation, the gap widens further. GPT-5-mini reaches a 73.2% win rate while LLaMA-3.3-70B falls to 7.0%. These results demonstrate that short, keyword-dense documents can succeed at retrieval, but downstream stages increasingly reward content that substantively addresses the query.

5.3 Insights & Exploration

In this section, we study effective strategy combinations and introduce practical optimization guidelines informed by our findings.

Strategy Combination. When applying SAGEO, multiple strategies can be combined. However, the All-in-One optimization results in Table 2 suggest that combining all strategies at once is not always effective. We therefore selectively evaluate combinations of top-performing strategies (Table 3). While these combinations improve

Table 4: Effectiveness of SAGEO strategies across different retrievers under the body-only setting. We report Hit Rate (H@20) and ΔRank for BM25, Dense, and Hybrid retrievers.

Strategy	BM25		Dense		Hybrid	
	H@20	ΔRank	H@20	ΔRank	H@20	ΔRank
Baseline	0.58 - - %	-	0.68 - - %	-	0.67 - - %	-
Auth.	0.57 -1%	-0.20	0.67 -1%	+0.10	0.66 -1%	-0.61
Cite	0.56 -3%	-1.50	0.66 -3%	-1.01	0.65 -4%	-2.80
EasyLang	0.52 -10%	-4.18	0.65 -4%	-2.20	0.62 -8%	-4.32
Fluency	0.57 -1%	-0.71	0.66 -2%	-0.46	0.65 -3%	-1.42
Quote	0.57 -1%	-0.33	0.66 -2%	-1.05	0.65 -2%	-1.68
Stats	0.57 -2%	-0.51	0.68 +1%	+0.76	0.66 -2%	-0.98
Tech.	0.50 -14%	-6.23	0.65 -4%	-1.85	0.60 -11%	-5.31
Unique	0.53 -8%	-3.47	0.66 -3%	-1.07	0.63 -7%	-3.20
All-in-One	0.50 -14%	-5.93	0.66 -2%	-1.26	0.62 -8%	-4.25
AutoGEO	0.37 -36%	-22.35	0.66 -2%	-1.49	0.63 -7%	-3.57
Avg.	0.53 -9%	-4.54	0.66 -2%	-0.95	0.64 -5%	-2.81

generation performance, retrieval rank degradation persists across all pairs. This indicates that naive pairing cannot simultaneously benefit all pipeline stages, motivating a stage-targeted approach.

Practical Guidelines. To this end, we introduce stage-aware optimization, which tailors optimization to the specific priorities of each pipeline stage. Our method builds on the following principles:

- **Enrich structural fields with key entities.** Core entities, numbers, and terms from body text are organized in structural fields to strengthen query-document matching at retrieval.
- **Make claims prominent and self-contained.** The main claim is placed at the start of the body, and each statement is ensured to carry specific evidence, increasing citability at generation.
- **Maintain topical coherence across paragraphs.** Ambiguous pronouns are replaced with explicit subject references, and core terms are naturally emphasized to keep paragraphs connected.
- **Adapt to domain context.** The document's domain is assessed before any changes, and both domain characteristics and user intent are considered to determine how to optimize.

We demonstrate the effectiveness of our method in Table 3, where it achieves competitive performance across all stages, with notable gains at both reranking and generation among all strategies evaluated. These results suggest that SAGEO Arena serves as an effective evaluation environment for developing practical SAGEO methods, enabling stage-level analysis and providing actionable insights. The detailed prompt used in our method is provided in Appendix E.

5.4 Supplementary Analysis

The primary objective of SAGEO Arena is to derive actionable findings that generalize to real-world SAGE. We therefore adopt BM25 as the first-stage retriever, the standard choice for web-scale retrieval given its scalability and efficiency, paired with representative models for reranking and generation. Nevertheless, we verify the robustness of our findings with alternative model choices, varying the retriever, reranker, and generator in the body-only setting.

Table 5: Results of SAGEO strategies under BM25 retrieval with alternative rerankers or generators in the body-only setting. Each evaluation varies one component independently, with the default model shown on the left within each block.

Strategy	Reranker				Generator			
	Qwen3-Reranker		BGE-Reranker-v2-m3		GPT-5-mini		Claude Sonnet	
	H@10	ΔRank	H@10	ΔRank	Cite	ΔRank	Cite	ΔRank
Baseline	1.00 - - %	-	1.00 - - %	-	0.50 - - %	-	0.77 - - %	-
Auth.	0.91 -9%	-0.24	0.91 -9%	-0.44	0.49 -3%	-0.10	0.75 -3%	-0.11
Cite	0.87 -13%	-0.45	0.85 -15%	-0.94	0.48 -4%	-0.13	0.69 -11%	-0.51
EasyLang	0.84 -16%	-0.54	0.78 -22%	-1.45	0.49 -4%	-0.09	0.67 -13%	-0.78
Fluency	0.91 -9%	-0.18	0.92 -8%	-0.31	0.50 -1%	-0.01	0.74 -4%	-0.14
Quote	0.90 -10%	-0.36	0.89 -11%	-0.69	0.47 -7%	-0.26	0.71 -8%	-0.45
Stats	0.90 -10%	-0.33	0.90 -10%	-0.54	0.48 -4%	-0.18	0.76 -1%	+0.03
Tech.	0.80 -20%	-1.03	0.69 -31%	-2.08	0.47 -6%	-0.15	0.67 -13%	-0.80
Unique	0.86 -14%	-0.70	0.81 -19%	-1.33	0.46 -8%	-0.33	0.68 -12%	-0.80
All-in-One	0.83 -17%	-0.68	0.73 -27%	-1.83	0.49 -2%	-0.03	0.69 -11%	-0.60
AutoGEO	0.58 -42%	-2.28	0.65 -35%	-2.17	0.39 -22%	-0.78	0.48 -38%	-1.94
Avg.	0.84 -16%	-0.68	0.81 -19%	-1.18	0.47 -6%	-0.21	0.68 -12%	-0.61

Robustness across Retrievers. We additionally evaluate two alternative settings: BGE-base-en-v1.5 [51] as a dense retriever, and its combination with BM25 as a hybrid retriever. As shown in Table 4, the trend is consistent across all three settings. Body-only optimization does not reliably improve retrieval visibility, and often lowers the target document's rank. This confirms that the observed degradation is not specific to BM25, but reflects a broader limitation of optimizing body text alone regardless of the underlying retriever.

Robustness across Rerankers and Generators. We further test two substitutions: replacing the reranker with BGE-Reranker-v2-m3 [5, 30] and replacing the generator with Claude-Sonnet-4.6. As shown in Table 5, the trend persists in both cases. Specifically, the reranker substitution moderately amplifies the degradation, indicating that lighter rerankers are more sensitive to body-only optimization. The generator substitution also worsens both Citation Rate and Rank Change for most strategies. Overall, these results confirm that body-only optimization can degrade visibility across the full pipeline, motivating stage-aware optimization in SAGE.

6 Conclusion

In this paper, we introduce SAGEO Arena, a realistic evaluation environment for stage-level visibility analysis of Search-Augmented Generative Engine Optimization. By integrating a full generative search pipeline with a large-scale corpus preserving structural information, SAGEO Arena enables investigation of how optimization signals propagate in generative search. We observe that findings from existing benchmarks do not readily generalize to realistic settings, and that stage-aware optimization targeting each pipeline stage is crucial for visibility in generative search. Overall, SAGEO Arena offers a reproducible environment for developing optimization strategies that generalize to real-world generative search.

Acknowledgments

This work was supported by the IITP grants funded by the Korea government (MSIT) (RS-2024-00457882, AI Research Hub Project; RS-2026-25520654).

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* (Barcelona, Spain) *(KDD '24)*. Association for Computing Machinery, New York, NY, USA, 5–16. doi:10.1145/3637528.3671900

[2] Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. 2018. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. arXiv:1611.09268 [cs.CL] https://arxiv.org/abs/1611.09268

[3] Vera Boteva, Demian Gholipour, Artem Sokolov, and Stefan Riezler. 2016. A Full-Text Learning to Rank Dataset for Medical Information Retrieval. In *Proceedings of the European Conference on Information Retrieval (ECIR)* (Padova, Italy). Springer.

[4] Jiawei Chen, Hongyu Lin, Xianpei Han, and Le Sun. 2024. Benchmarking large language models in retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 17754–17762.

[5] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. BGE M3-Embedding: Multi-Lingual, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. arXiv:2402.03216 [cs.CL]

[6] Mingyang Chen, Tianpeng Li, Haoze Sun, Yijie Zhou, Chenzheng Zhu, Haofen Wang, Jeff Z. Pan, Wen Zhang, Huajun Chen, Fan Yang, Zenan Zhou, and Weipeng Chen. 2025. ReSearch: Learning to Reason with Search for LLMs via Reinforcement Learning. arXiv:2503.19470 [cs.AI] https://arxiv.org/abs/2503.19470

[7] Mahe Chen, Xiaoxuan Wang, Kaiwen Chen, and Nick Koudas. 2025. Generative engine optimization: How to dominate ai search. *arXiv preprint arXiv:2509.08919* (2025).

[8] Mahe Chen, Xiaoxuan Wang, Kaiwen Chen, and Nick Koudas. 2026. Navigating the Shift: A Comparative Analysis of Web Search and Generative AI Response Generation. *arXiv preprint arXiv:2601.16858* (2026).

[9] Qiyuan Chen, Jiahe Chen, Hongsen Huang, Qian Shao, Jintai Chen, Renjie Hua, Hongxia Xu, Ruijia Wu, Ren Chuan, and Jian Wu. 2025. Beyond Keywords: Driving Generative Search Engine Optimization with Content-Centric Agents. *arXiv preprint arXiv:2509.05607* (2025).

[10] Xiaolu Chen, Haojie Wu, Jie Bao, Zhen Chen, Yong Liao, and Hu Huang. 2025. Role-Augmented Intent-Driven Generative Search Engine Optimization. *arXiv preprint arXiv:2508.11158* (2025).

[11] Xiang Deng, Prashant Shiralkar, Colin Lockard, Binxuan Huang, and Huan Sun. 2022. DOM-LM: Learning Generalizable Representations for HTML Documents. *ArXiv* abs/2201.10608 (2022). https://api.semanticscholar.org/CorpusID:246285527

[12] Elastic. 2025. Relevance Tuning Guide, Weights and Boosts. https://www.elastic.co/guide/en/app-search/current/relevance-tuning-guide.html. Accessed: 2026-01-08.

[13] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997* 2, 1 (2023).

[14] Google. 2024. How Google Search Works. https://developers.google.com/search/docs/fundamentals/how-search-works. Accessed: 2025-01-19.

[15] Google. 2025. SEO Starter Guide. https://developers.google.com/search/docs/fundamentals/seo-starter-guide. Accessed: 2026-01-17.

[16] Google Cloud. 2026. Vertex AI RAG Engine Overview. https://docs.cloud.google.com/vertex-ai/generative-ai/docs/rag-engine/rag-overview. Accessed: 2026-01-07.

[17] Gregory Goren, Oren Kurland, Moshe Tennenholtz, and Fiana Raiber. 2020. Ranking-incentivized quality preserving content modification. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 259–268.

[18] Venkat N Gudivada, Dhana Rao, and Jordan Paris. 2015. Understanding search-engine optimization. *Computer* 48, 10 (2015), 43–52.

[19] Yu Guo, Zhengyi Ma, Jiaxin Mao, Hongjin Qian, Xinyu Zhang, Hao Jiang, Zhao Cao, and Zhicheng Dou. 2022. Webformer: Pre-training with Web Pages for Information Retrieval. *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (2022). https://api.semanticscholar.org/CorpusID:250340272

[20] Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *arXiv preprint arXiv:2503.09516* (2025).

[21] Zixuan Ke, Weize Kong, Cheng Li, Mingyang Zhang, Qiaozhu Mei, and Michael Bendersky. 2024. Bridging the preference gap between retrievers and llms. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 10438–10451.

[22] Hyunseo Kim, Sangam Lee, Kwangwook Seo, and Dongha Lee. 2025. BESPOKE: Benchmark for Search-Augmented Large Language Model Personalization via Diagnostic Feedback. *arXiv preprint arXiv:2509.21106* (2025).

[23] Sunghwan Kim, Ryang Heo, Yongsik Seo, Jinyoung Yeo, and Dongha Lee. 2026. AgenticShop: Benchmarking Agentic Product Curation for Personalized Web Shopping. In *Proceedings of the ACM Web Conference 2026* (United Arab Emirates) *(WWW '26)*. Association for Computing Machinery, New York, NY, USA, 2489–2500. doi:10.1145/3774904.3792724

[24] Aounon Kumar and Himabindu Lakkaraju. 2024. Manipulating large language models to increase product visibility. *arXiv preprint arXiv:2404.07981* (2024).

[25] Arlen Kumar and Leanid Palkhouski. 2025. AI Answer Engine Citation Behavior An Empirical Analysis of the GEO16 Framework. *arXiv preprint arXiv:2509.10762* (2025).

[26] Oren Kurland and Moshe Tennenholtz. 2022. Competitive search. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2838–2849.

[27] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: A Benchmark for Question Answering Research. *Transactions of the Association for Computational Linguistics* 7 (2019), 452–466. doi:10.1162/tacl_a_00276

[28] Dirk Lewandowski, Sebastian Sünkler, and Nurce Yagci. 2021. The influence of search engine optimization on Google's results: A multi-dimensional approach for detecting SEO. In *Proceedings of the 13th ACM Web Science Conference 2021* (Virtual Event, United Kingdom) *(WebSci '21)*. Association for Computing Machinery, New York, NY, USA, 12–20. doi:10.1145/3447535.3462479

[29] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems* 33 (2020), 9459–9474.

[30] Chaofan Li, Zheng Liu, Shitao Xiao, and Yingxia Shao. 2023. Making Large Language Models A Better Foundation For Dense Retrieval. arXiv:2312.15503 [cs.CL]

[31] Xiaoxi Li, Jiajie Jin, Guanting Dong, Hongjin Qian, Yongkang Wu, Ji-Rong Wen, Yutao Zhu, and Zhicheng Dou. 2026. Webthinker: Empowering large reasoning models with deep research capability. *Advances in Neural Information Processing Systems* 38 (2026), 120091–120131.

[32] Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. WWW'18 Open Challenge: Financial Opinion Mining and Question Answering. (2018), 1941–1942.

[33] Microsoft. 2025. How Bing Delivers Search Results. https://support.microsoft.com/en-us/bing/how-bing-delivers-search-results. Accessed: 2026-01-19.

[34] Microsoft. 2025. Webmasters Guidelines. https://www.bing.com/webmasters/help/webmasters-guidelines-30fba23a. Accessed: 2026-01-19.

[35] Microsoft. 2026. Copilot Search in Bing. https://www.microsoft.com/en-us/bing/copilot-search/. Accessed: 2026-04-11.

[36] Fredrik Nestaas, Edoardo Debenedetti, and Florian Tramèr. 2025. Adversarial search engine optimization for large language models. In *International Conference on Learning Representations*, Vol. 2025. 4857–4888.

[37] 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*. 9523–9552.

[38] Haritz Puerto, Martin Gubri, Tommaso Green, Seong Joon Oh, and Sangdoo Yun. 2025. C-SEO Bench: Does Conversational SEO Work? *arXiv preprint arXiv:2506.11097* (2025).

[39] Nimrod Raifer, Fiana Raiber, Moshe Tennenholtz, and Oren Kurland. 2017. Information Retrieval Meets Game Theory: The Ranking Competition Between Documents' Authors. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Shinjuku, Tokyo, Japan) *(SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 465–474. doi:10.1145/3077136.3080785

[40] David Rau, Hervé Déjean, Nadezhda Chirkova, Thibault Formal, Shuai Wang, Stéphane Clinchant, and Vassilina Nikoulina. 2024. Bergen: A benchmarking library for retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*. 7640–7663.

[41] Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.* 3, 4 (April 2009), 333–389. doi:10.1561/1500000019

[42] Stephen E. Robertson, Hugo Zaragoza, and Michael J. Taylor. 2004. Simple BM25 extension to multiple weighted fields. In *International Conference on Information and Knowledge Management*. https://api.semanticscholar.org/CorpusID:16628332

[43] Dushyant Sharma, Rishabh Shukla, Anil Kumar Giri, and Sumit Kumar. 2019. A brief review on search engine optimization. In *2019 9th international conference on cloud computing, data science & engineering (confluence)*. IEEE, 687–692.

[44] Lakshay Sharma, Laura Graesser, Nikita Nangia, and Utku Evci. 2019. Natural Language Understanding with the Quora Question Pairs Dataset. arXiv:1907.01041 [cs.CL] https://arxiv.org/abs/1907.01041

[45] Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024. Replug: Retrieval-augmented black-box language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. 8371–8384.

[46] Jiejun Tan, Zhicheng Dou, Wen Wang, Mang Wang, Weipeng Chen, and Ji-Rong Wen. 2025. HtmlRAG: HTML is Better Than Plain Text for Modeling Retrieved Knowledge in RAG Systems. In *Proceedings of the ACM on Web Conference 2025* (Sydney NSW, Australia) *(WWW '25)*. Association for Computing Machinery, New York, NY, USA, 1733–1746. doi:10.1145/3696410.3714546

[47] Erdinç Uzun, Hayri Volkan Agun, and TarıK Yerlikaya. 2013. A hybrid approach for extracting informative content from web pages. *Inf. Process. Manage.* 49, 4 (July 2013), 928–944. doi:10.1016/j.ipm.2013.02.005

[48] Qifan Wang, Yi Fang, Anirudh Ravula, Fuli Feng, Xiaojun Quan, and Dongfang Liu. 2022. WebFormer: The Web-page Transformer for Structure Information Extraction. In *Proceedings of the ACM Web Conference 2022* (Virtual Event, Lyon, France) *(WWW '22)*. Association for Computing Machinery, New York, NY, USA, 3124–3133. doi:10.1145/3485447.3512032

[49] Xiaohua Wang, Zhenghua Wang, Xuan Gao, Feiran Zhang, Yixin Wu, Zhibo Xu, Tianyuan Shi, Zhengyuan Wang, Shizheng Li, Qi Qian, et al. 2024. Searching for best practices in retrieval-augmented generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. 17716–17736.

[50] Yujiang Wu, Shanshan Zhong, Yubin Kim, and Chenyan Xiong. 2025. What Generative Search Engines Like and How to Optimize Web Content Cooperatively. *arXiv preprint arXiv:2510.11438* (2025).

[51] Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-Pack: Packaged Resources To Advance General Chinese Embedding. arXiv:2309.07597 [cs.CL]

[52] Rongwu Xu, Xuan Qi, Zehan Qi, Wei Xu, and Zhijiang Guo. 2024. DebateQA: Evaluating Question Answering on Debatable Knowledge. arXiv:2408.01419 [cs.CL] https://arxiv.org/abs/2408.01419

[53] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*. 2369–2380.

[54] Lan Yi, Bing Liu, and Xiaoli Li. 2003. Eliminating noisy information in Web pages for data mining. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (Washington, D.C.) *(KDD '03)*. Association for Computing Machinery, New York, NY, USA, 296–305. doi:10.1145/956750.956785

[55] Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models. *arXiv preprint arXiv:2506.05176* (2025).

[56] Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei Liu. 2025. Deepresearcher: Scaling deep research via reinforcement learning in real-world environments. *arXiv preprint arXiv:2504.03160* (2025).

A Benchmark Construction

Statistics. SAGEO Arena comprises nine domains sourced from established query datasets: MS MARCO [2] (Web Search), Natural Questions (General QA) [27], HotpotQA [53] (Multi-hop QA), NFCorpus [3] (Biomedical), Quora [44] (Community QA), FiQA [32] (Finance), DebateQA [52] (Debate), E-commerce (Shopping), and Researchy (Academic) [50]. Domain selection ensures diversity in content characteristics, query complexity, and information-seeking behaviors. Detailed statistics are presented in Table 6.

Table 6: Benchmark statistics across nine domains.

Source Dataset	Domain	#Sampled Queries	#Retrieved Docs
MS MARCO	Web Search	300	21,880
Natural Questions	General QA	300	21,921
HotpotQA	Multi-hop QA	300	13,409
NFCorpus	Biomedical	300	21,079
Quora	Community QA	300	21,734
FiQA	Finance	300	16,771
DebateQA	Debate	300	17,443
E-commerce	Shopping	300	18,885
Researchy	Academic	300	17,881
Total		**2,700**	**171,003**

Difference from RAG Benchmarks. Standard RAG benchmarks evaluate whether a system generates faithful and relevant answers given a query and a document corpus. While this setting is useful for measuring answer quality, it does not directly capture the content creator's perspective, where the central question is whether a specific document can remain visible across retrieval, reranking, and generation. SAGEO Arena addresses this gap by approximating the real-world generative search pipeline and measuring target document visibility before and after optimization. Moreover, SAGEO Arena preserves structural information from web documents. These fields provide useful signals for how web documents are interpreted and surfaced in search systems, but are typically discarded in RAG-style benchmarks that operate mainly on plain body text passages.

B Implementation Details

We instantiate the pipeline described in Section 3.3 with the following configuration. We segment document body text into passages of 256 tokens with a 64-token overlap. During indexing, we assign equal weights across structural fields and chunked passages to establish a controlled and fair baseline. In practice, search engines apply varying weights to different fields [12]; however, adopting uniform weights ensures that observed visibility changes stem from optimization strategies rather than field-specific biases. We retrieve the top-100 passages per query and rescore them using Qwen3-Reranker-4B [55]. For generation, the top-10 candidates are retained for response synthesis, using GPT-5-mini as the default generation model. For each test query, we first establish a baseline by running the query through the full search pipeline, then randomly select a target document from the top-10 candidates that enter the generation stage (i.e., ranked within the top-10 after reranking), ensuring sufficient relevance to the test query. All SAGEO optimization strategies are implemented using GPT-5-mini with strategy-specific prompts following [1] for the eight content modification strategies, [38] for All-in-One, and [50] for AutoGEO.

C Further Analysis

Optimization strategies show limited robustness across query formulations. Real users express information needs through diverse query formulations, making robustness to query variation essential for effective optimization. To examine this, we evaluate optimized documents against four query rewriting strategies: expansion (adding specificity), simplification (keeping core keywords), rephrasing (lexical substitution), and abstraction (semantic generalization). We report results averaged across eight optimization strategies from [1] in Figure 7. We observe consistent visibility degradation under all query variations, with a progressive drop as queries deviate further from the original phrasing. Since document optimization is performed without knowledge of the incoming query, we find that it tends to elaborate on the document's existing content, making the document more specific to its original topic and vocabulary. As a result, optimization effectiveness becomes sensitive to query formulation, especially when rewritten queries reduce lexical overlap with the optimized document. For example, the degradation is most pronounced under abstraction, where queries are generalized beyond the document's original wording, causing the most severe drop. These results indicate that current strategies show limited transferability, motivating robust SAGEO methods that preserve visibility across diverse query formulations.

Table 7: The prompt template for stage-aware SAGEO.

Stage-Aware SAGEO Prompt

[Task Description]
Optimize the following document with these strategies. Keep the content faithful to the original.

[Pre-Optimization Considerations]
Before optimizing, think about two things:
1. **Domain**: Consider what domain this document belongs to (e.g., medical, finance, e-commerce, technical, casual). Match the tone and vocabulary to what readers in that domain expect.
2. **Quality**: If a field is already clear, specific, and well-written, keep it as-is. Only optimize fields that genuinely benefit from it. Not every document needs heavy changes.

[Optimization Strategies]
1. Entity mirroring (structural fields):
Incorporate key entities, numbers, and domain terms from the body into the title, meta_description, headings, and jsonld_text. Add a keyword-rich summary sentence while keeping it compact. Skip if the structural fields already contain the right keywords.

2. Fluent, easy language (all text):
Rewrite sentences to be smooth, clear, and easy to read. Use simple words and short sentences. Avoid jargon when a plain alternative exists. If the writing is already clear and fluent, leave it unchanged.

3. Concrete evidence (body text):
Make claims specific. Bring front the main claim to the very start of the body. Each claim should be self-contained — a reader should understand it without reading surrounding text. If claims are already specific, do not rephrase them.

4. Keyword reinforcement (body text):
Naturally repeat the document's core topic terms and key phrases throughout the body. Use the main subject name instead of pronouns where it reads naturally. This keeps every paragraph clearly connected to the topic.

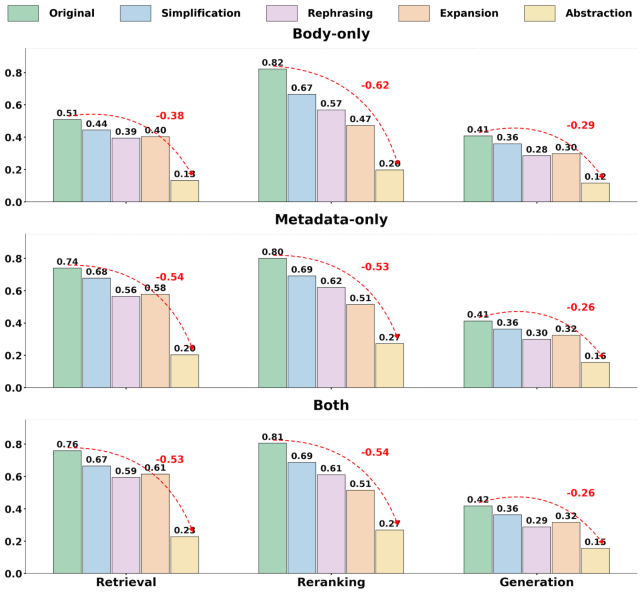


Figure 7: Document visibility across pipeline stages under four query variations for three optimization scopes. Values show mean Hit Rate at retrieval and reranking, and mean Citation Rate at generation, averaged across eight optimization strategies. Red numbers indicate percentage point change from the original query to the lowest-performing variation.

Answer Quality Analysis. Although our primary focus is to evaluate document visibility, we provide additional analysis on SAGEO's impact on answer quality. SAGEO Arena studies white-hat document optimization, where methods cooperatively enhance content rather than inject adversarial signals. Moreover, each generated answer is grounded in multiple retrieved documents, so modifying one target document is unlikely to meaningfully change overall answer quality. To empirically validate this, we conduct a sanity check using 3,000 before-and-after optimization samples, evaluating whether generated answers address the query's information need using GPT-5.4 as the evaluator. We observe only a small change after optimization across all methods, from 88.9% to 87.3%.

D Limitations

While SAGEO Arena approximates a realistic generative search pipeline to provide actionable insights, it does not fully reproduce commercial search engines with proprietary ranking signals or user behavior signals. Since these signals are difficult to access and reproduce, SAGEO Arena focuses on controllable factors that content creators can observe and directly modify to improve document visibility. Nevertheless, incorporating richer ranking and user behavior signals remains an important direction for future SAGEO research.

E Prompt for StageAware SAGEO

Based on the empirical findings from SAGEO Arena, we introduce a stage-aware SAGEO strategy that tailors optimization to the specific priorities of each pipeline stage. The prompt is shown in Table 7.