EVIREPORT: From Reasoned Outlines to Evidence Tracked Long-Form Reports

Anonymous ACL submission

Abstract

Evidence-intensive analytical reports are expected to be fact-dense, quantitatively correct, and supported by figures. Yet one-shot long-form generation with large language models (LLMs) frequently produces fluent but under-supported drafts: core facts are missed, numbers drift, and key visuals are absent, making the report hard to trust. We propose EVIREPORT, an evidence-tracked report-writing workflow that improves reliability by (i) organizing corpus evidence into compact, traceable units and retrieves query-relevant subgraphs into retrieval-ready packages (ii) leveraging a reasoning-focused LLM sketches a high-level plan for full coverage, then a chat-based LLM sharpens it into a detailed hierarchical outline with explicit scope and ordering (iii) rive generation with a facts-first iterative loop: extracting verifiable facts, composing strictly from those facts, then triggering gap-aware append queries to fill missing evidence To evaluate both correctness and completeness, we introduce EviReportBench, a benchmark instantiated on data-rich indicator reports that measures factual accuracy (claim verification), factual coverage (quiz-based evaluation), and visual evidence integration (image recall). Across 8 topics, experiments show that EVIREPORT consistently outperforms strong baselines in factual coverage (2.16×), factual accuracy (+8.9 points), and visual evidence integration (+34 points), approaching the quality of expert-written reports across multiple dimensions.

1 Introduction

Evidence-intensive *long-form* reports are widely used for assessment and policy decision-making because they synthesize heterogeneous materials (papers, official documents, indicator tables, and figures) into a coherent narrative with verifiable claims. A representative example is the **United Nations Sustainable Development Goals (SDGs)** reporting ecosystem: SDGs define 17 goals with an official global indicator framework, and progress is regularly documented through indicator-centric reports that integrate multi-source evidence and rely heavily on figures (United Nations Statistics Division; United Nations General Assembly, 2017; United Nations Department of Economic and Social Affairs, 2025). In this work, we focus on automating SDG-style report writing as a primary application scenario and a stress test for evidence-intensive report generation.

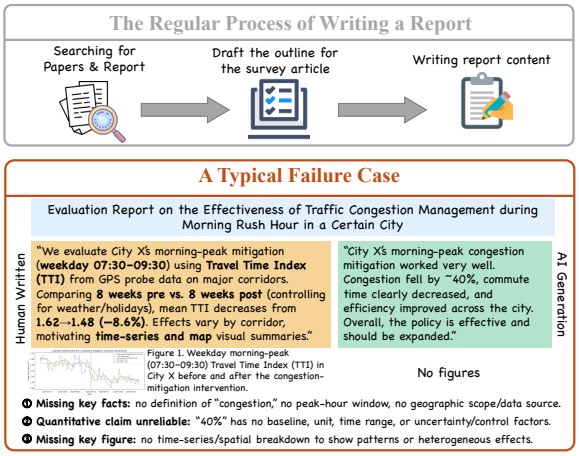


Figure 1: Report writing is evidence-intensive. **Top:** a regular workflow (search, outline, draft). **Bottom:** a typical failure of direct long-form LLM generation on a morning-peak congestion mitigation report: the AI text often (1) misses key facts, (2) makes unverifiable numbers, and (3) lacks critical figures, compared with a human-written version grounded in metric/time window/scope and a time-series plot.

Large language models (LLMs) can accelerate drafting (Jin et al., 2024; OpenAI, 2025a), but directly prompting an LLM to generate a full report often leads to three recurring failures (Fig. 1). First, the output may exhibit *low factual coverage*: it misses topic-critical facts while including generic or irrelevant points that do not advance the analysis. Second, the output may contain *quantitative unreliability*, including factual errors in

trends/ranges/numbers and under-specified claims that omit essential constraints such as year, unit, population, or data source. Third, it often shows weak visual evidence integration, omitting key figures (e.g., maps and trend plots) or failing to align textual claims with the corresponding visuals. These issues directly undermine report credibility and motivate workflows that explicitly control evidence use, planning, and multimodal grounding.

Evidence-intensive long-form report generation poses challenges beyond producing fluent text. Real-world report evidence is typically *structured*: high-level targets/indicators are argued through concrete cases supported by specific datasets and analytical methods, and ignoring such evidence chains leads to omissions or unsupported statements. Report quality also depends on *planning*: outlines determine what will be covered and how the narrative is organized, yet producing comprehensive and non-redundant outlines under long contexts and complex constraints remains difficult. Moreover, many reports are *multimodal*, where figures carry core information, and evaluation is non-trivial because valid reports are not unique and single-aspect scoring cannot diagnose whether a system is correct, complete, and visually grounded.

To address these challenges, we present EVIREPORT, an end-to-end workflow for evidence-tracked multimodal long-form report generation. EVIREPORT follows a retrieve–plan–write paradigm with three stages. (1) **Graph-enhanced evidence retrieval**: we build an evidence-oriented knowledge structure from the reference corpus to organize evidence into compact, traceable units, retrieve and prune query-relevant subgraphs, and summarize associated evidence into retrieval-ready packages. (2) **Two-stage outline planning**: a reasoning-oriented LLM drafts a coarse plan to ensure global coverage, and a chat-oriented LLM refines it into a fine-grained hierarchical outline with clearer scopes and ordering. (3) **Evidence-guided multimodal writing**: for each subsection, EVIREPORT performs targeted text and figure retrieval (including caption-guided image retrieval) and generates content via a three-step iterative loop (facts $\rightarrow$ content $\rightarrow$ gap-aware append query) to reduce under-specified claims and recover missing evidence.

A major obstacle to progress is the lack of a diagnostic benchmark that jointly measures *correctness*, *coverage*, and *visual evidence integration* for real-world, data- and figure-intensive reports. To fill this gap, we introduce **EviReportBench**, a multi-dimensional benchmark instantiated on SDG-style indicator reports (United Nations Department of Economic and Social Affairs, 2025; United Nations Secretary-General, 2025). Beyond claim-level factual verification, EviReportBench measures factual coverage via a quiz-based protocol derived from authoritative reports, and evaluates multimodal integration via image recall against topic-specific ground-truth figures. Experiments across eight representative topics show that EVIREPORT consistently outperforms strong baselines on factual accuracy, factual coverage, and image recall. **Evaluation code.** We release the benchmark construction and scoring code at `https://anonymous.4open.science/r/EviReport-03E2`.

In summary, our contributions are:

- We propose EVIREPORT, a workflow for evidence-tracked multimodal long-form report generation with graph-enhanced retrieval, two-stage outline planning, and iterative evidence-guided writing.
- We introduce caption-guided image retrieval and a three-step iterative writing loop to improve quantitative reliability, factual coverage, and visual evidence integration.
- We build EviReportBench, a new multi-dimensional benchmark instantiated on SDG-style indicator reports, with claim verification for factual accuracy, quiz-based evaluation for factual coverage, and image recall for multimodal evidence integration.

2 Related Work

Long-form report generation, planning, grounding, and verification. Although LLMs excel at short-form NLG, generating long-form reports that are well-structured, coherent, and comprehensive remains challenging under long-context constraints and complex instructions (Dong et al., 2024; Kumar et al., 2024). Prior work therefore adopts planning and multi-stage pipelines that separate global organization from local drafting (Tang et al., 2022; Tan et al., 2021); for example, STORM (Shao et al., 2024) introduces a pre-writing stage to research a topic and construct an outline before drafting Wikipedia-style articles, while AutoSurvey (Wang et al., 2024) decomposes survey writing into retrieval, outlining, subsection drafting, refinement, and evaluation. Complementary to

planning, verification-based prompting improves factuality by checking and revising model outputs (Gao et al., 2023a; Zhao et al., 2023; Madaan et al., 2023); Chain-of-Verification (CoVe) generates verification questions and revises drafts accordingly (Dhuliawala et al., 2024). On the evaluation side, claim-level verification (e.g., FactScore-style protocols) provides more objective signals than holistic ratings (Min et al., 2023). Our work complements these directions with an iterative subsection-writing loop (facts → draft → gap-aware evidence expansion) and multi-dimensional evaluation.

Retrieval augmentation, graph-enhanced retrieval, and SDG-style reports. Retrieval augmentation grounds generation on external evidence (Lewis et al., 2020; Fan et al., 2024; Gao et al., 2023b), and structured/relational evidence motivates graph-based variants that retrieve entity-centric subgraphs for more complete evidence aggregation (Edge et al., 2024; Hu et al., 2024; Gao et al., 2025). Representative systems include KERAG (Sun et al., 2025), SubgraphRAG (Li et al., 2025), and KAG (Liang et al., 2024), with surveys summarizing common GraphRAG pipelines (Peng et al., 2024). The UN SDGs define 17 goals with an official indicator framework for monitoring progress (United Nations Statistics Division; United Nations General Assembly, 2017); SDG-style indicator reports are evidence- and data-intensive, contain dense quantitative claims, and rely heavily on figures (e.g., maps and trend plots), making them a strong stress test for grounded multimodal report generation. We therefore instantiate our benchmark on authoritative SDG-style reports to enable diagnostic evaluation of factual accuracy, factual coverage, and visual evidence integration.

3 Methodology

In this section, we propose EVIREPORT, an evidence-grounded workflow for long-form report generation with LLMs. As shown in Fig. 2, EVIREPORT consists of three main stages: knowledge-guided retrieval, outline planning, and content drafting. The retrieval stage constructs a lightweight report knowledge graph from the reference corpus and performs graph-enhanced retrieval to produce compact, traceable evidence packages. The outline planning stage adopts a coarse-to-fine strategy, where a reasoning-oriented LLM generates a high-level plan and a chat-based LLM refines it into a subsection-level outline. The content drafting stage conducts plan-conditioned multimodal retrieval (including caption-guided image retrieval) and generates each subsection via an iterative refinement loop to improve correctness, coherence, and completeness. Finally, we introduce EviReportBench, a multi-dimensional benchmark instantiated on SDG-style indicator reports to evaluate factual accuracy, factual coverage, and visual evidence integration. The details are elaborated in Sec. 3.2, Sec. 3.3, Sec. 3.4, and Sec. 3.5, respectively.

3.1 Preparation Phase

We preprocess heterogeneous reference materials to support evidence-grounded long-form report generation. Our sources include domain-specific research papers and governmental/organizational reports. Textual resources are parsed into a unified Markdown (MD) format, while structured indicator or statistics tables are cleaned and stored in a database. Since the end-to-end data pipeline (collection, parsing, chunking, and indexing) is described in Appendix A.2, we focus on how these resources are organized into retrieval-ready knowledge structures for downstream generation.

Reference preprocessing has two components: (i) constructing an evidence-oriented knowledge structure, and (ii) building retrieval corpora for text and images.

3.1.1 Evidence-Oriented Knowledge Structure

Analytical reports in many domains are typically organized around a stable *reporting framework* (e.g., a taxonomy of targets and measurable indicators) and are supported by case-based analyses grounded in specific datasets and computational methods. Motivated by this pattern, we build an evidence-oriented knowledge graph that links framework items (targets/indicators) to *cases*, and further connects each case to its *datasets* and *methods*, forming an explicit "indicator–case–dataset–method" evidence chain. This structure enables the system to retrieve compact and traceable evidence packages for a target report item, rather than relying on isolated text chunks. Appendix A.3 details the domain-specific instantiation of this graph, including schema design, extraction/fusion, and an example subgraph.

During report writing, the system retrieves case-level subgraphs and uses the linked datasets and methods to steer evidence-aware retrieval and generation. This design improves factual coverage and

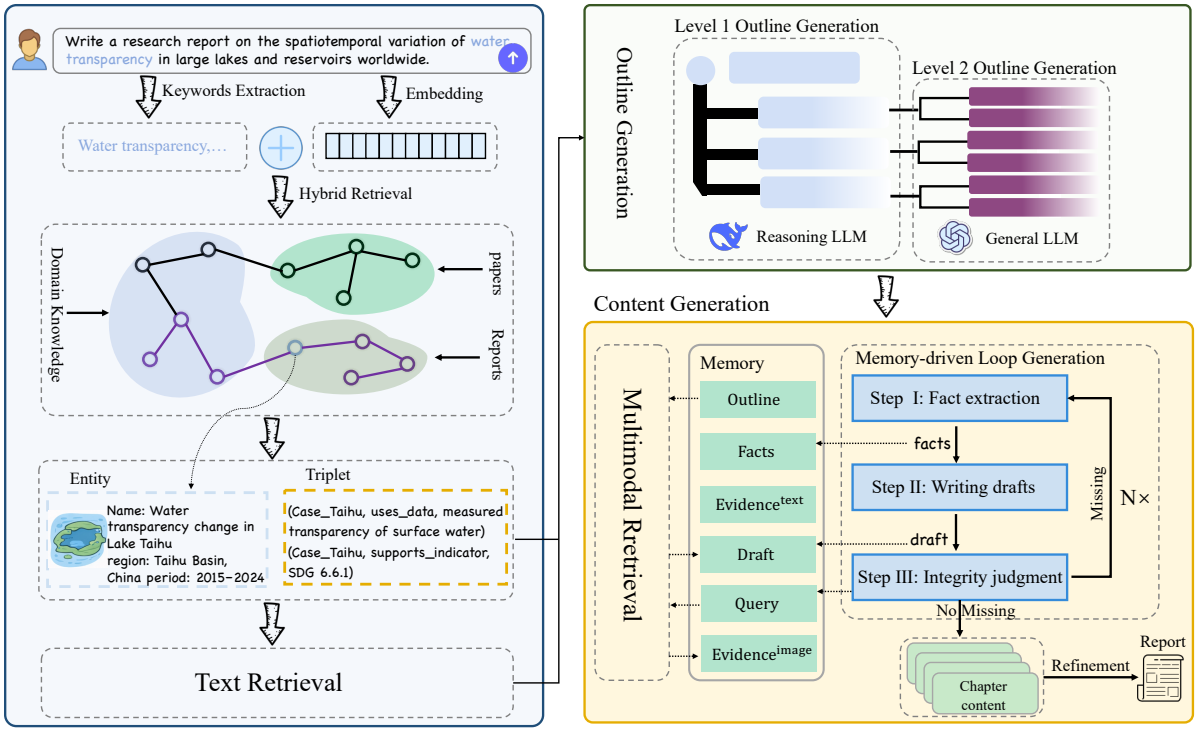


Figure 2: An overview of EVIREPORTfor evidence-grounded long-form report generation. EVIREPORTcombines (i) graph-enhanced hybrid retrieval over heterogeneous evidence to recall subgraphs/entities/triples for RAG, (ii) two-stage hierarchical outline planning, and (iii) an iterative generation loop that extracts facts from recalled context, writes drafts, and generates append queries when information is missing, producing structured reports with traceable evidence.

reduces unsupported statements by keeping generation tightly coupled with retrievable evidence.

3.1.2 Retrieval Corpus Construction

Based on the constructed knowledge structure, we build retrieval corpora for report generation. Parsed MD documents are segmented into chapter-aware semantic units and normalized with source and temporal metadata, enabling hybrid sparse–dense retrieval (Appendix A.2). In parallel, we construct an image retrieval database aligned with the text corpus: each figure is paired with its caption and surrounding context, and a vision–language model extracts lightweight metadata (keywords and semantic summaries) for indexing (Fig. 4).

At inference time, retrieval is conditioned on chapter- and subsection-level writing plans. Text evidence and image candidates are jointly matched to the target framework item (and its associated cases), allowing the system to recall multimodal evidence that is both relevant and traceable.

3.2 Knowledge-graph-enhanced RAG retrieval

To provide high-quality and verifiable evidence prior to planning and drafting, we introduce a graph-enhanced retrieval-augmented generation (KG-RAG) module (Algorithm 1). Given an input query q, a knowledge graph $\mathcal{G}$ constructed from the reference corpus, and a text database $\mathcal{D}$, the module outputs a set of evidence pairs $\mathcal{P}$. Each pair consists of a compact, query-relevant subgraph and a corresponding textual summary, serving as *retrieval-ready evidence packages* for downstream outline generation and subsection writing.

Concretely, the system first extracts keywords from q and computes a semantic embedding to support hybrid retrieval. It then retrieves candidate *case nodes* from $\mathcal{G}$—case nodes represent concrete evidence units (e.g., empirical studies, datasets, or documented analyses) that can anchor report claims. After merging and deduplicating candidates, we keep the top-K nodes to form $\mathcal{E}$. For each $c \in \mathcal{E}$, we extract an entity-centered subgraph $\mathcal{G}_c$ within h hops and prune it to obtain $\mathcal{G}'_c$, retaining only nodes and relations that are relevant to

the query and useful for report writing (e.g., linked entities, data sources, and methods).

Next, the pruned subgraph $\mathcal{G}'_c$ is converted into retrieval constraints (entities/relations) to query the text database $\mathcal{D}$, yielding evidence text blocks $\mathcal{B}_c$ that are semantically consistent with the case and its analytical context. We summarize $\mathcal{B}_c$ into s_c to distill key facts, numbers, and explicit gaps conditioned on q, and then augment the subgraph with summary-derived attributes to obtain $\mathcal{G}''_c$. Finally, we collect paired evidence $(\mathcal{G}''_c, s_c)$ across all candidates to form $\mathcal{P}$, which provides compact, traceable inputs for the subsequent *plan–write* stages.

3.3 Two-stage report outline generation

A high-quality outline is the backbone of long-form report writing: it fixes the document structure, allocates space across themes, and implicitly determines what evidence will be retrieved and discussed. In our setting, outlining must satisfy two competing requirements: it should be *globally coherent* and cover major analytical dimensions, while remaining *grounded* in the available evidence under long-context constraints. In practice, asking a single LLM to directly produce a detailed outline from lengthy, heterogeneous evidence often yields either a generic template with low evidence coverage or an overfit, fragmented structure.

Design insight. We find that outline generation is best treated as a *planning problem* rather than a one-shot formatting task. Global structure ("what to cover") and local organization ("how to break down and order content") require different behaviors: the former benefits from deliberate reasoning over evidence and reporting conventions, while the latter benefits from instruction-following and stylistic consistency. This motivates a coarse-to-fine decomposition.

Two-stage planning. We adopt a two-stage outline generation strategy. In the first stage, a reasoning-oriented LLM produces a *coarse-grained chapter plan* that emphasizes global structure and major analytical sections (e.g., background, data/measurement, empirical findings, comparisons, limitations, and implications), conditioned on the retrieved evidence packages. In the second stage, a chat-based LLM expands and refines the plan into a *fine-grained, subsection-level outline*, ensuring consistent formatting, logical ordering, and improved coverage by explicitly mapping each subsection to a small set of supporting evidence packages. By separating global planning from local decomposition, this design yields outlines that are both structurally coherent and evidence-aware, providing stable writing plans for the downstream subsection drafting stage.

3.4 Content generation

Given the refined outline $\mathcal{O} = \{o_1, \ldots, o_N\}$ (Sec. 3.3), EVIREPORT drafts the report in a *plan-conditioned, serial* manner. We generate subsections sequentially so that each subsection o_i can condition on previously generated content $\mathcal{C}_{<i} = \{c_1, \ldots, c_{i-1}\}$, helping maintain global consistency in terminology, narrative flow, and cross-section references. As illustrated in Fig. 2 and formalized in Algorithm 2, each subsection is produced through an iterative *retrieve–generate* loop with explicit multimodal grounding.

Plan-conditioned multimodal retrieval. For each subsection o_i, we construct a text retrieval query q_i^{text} and an image retrieval query q_i^{img} from $(o_i, \mathcal{C}_{<i})$. We retrieve text evidence blocks $\mathcal{B}_i$ from the text database $\mathcal{D}_t$ and image candidates $\mathcal{I}_i^{(0)}$ from the image database $\mathcal{D}_v$. To better align visual evidence with the subsection content, we additionally extract caption-like strings $\mathcal{C}_i^{\text{cap}}$ from $\mathcal{B}_i$ (e.g., via lightweight pattern rules) and use them as auxiliary queries for caption-based image retrieval, producing $\mathcal{I}_i^{(1)}$. We then merge and deduplicate images to obtain $\mathcal{I}_i$, and pack their metadata (caption, source, and brief description) as model-readable context.

Three-step iterative writing loop. We structure subsection writing as an explicit three-step loop that separates *fact extraction*, *content drafting*, and *gap-aware evidence expansion*. Concretely, given the assembled context $\mathcal{X}_i \leftarrow [\tilde{\mathcal{B}}_i; \tilde{\mathcal{I}}_i; \mathcal{C}_{<i}]$, the model first extracts a fact list F_i grounded in $\mathcal{X}_i$ (Step 1), then drafts subsection content by organizing and verbalizing F_i under the plan o_i (Step 2), and finally identifies missing items and generates an append query q_i^{app} to retrieve additional evidence (Step 3). The newly retrieved evidence is appended to the context and the loop repeats, progressively reducing omissions and weakening unsupported statements. In our main experiments, we set the number of iterations to $T{=}3$; we report results under different iteration budgets in the Appendix. A.5.1. After the loop terminates, we finalize the subsection content c_i using the augmented context. After all subsections are generated, we perform a lightweight refinement pass to improve style consistency, pro-

ducing the final multimodal report $\mathcal{R}$.

3.5 Evaluation Benchmark

Evaluating AI-generated long-form reports is challenging: many outputs can be acceptable for the same prompt, and holistic human ratings are often subjective and poorly attributable to specific system components. To enable reproducible and diagnostic evaluation aligned with the failure modes we target (hallucinated facts, low factual coverage, and missing visual evidence), we introduce **EviReportBench**, a benchmark for *evidence-grounded multimodal report generation*.

Why SDG-style indicator reports? We instantiate EviReportBench on SDG-style analytical reports not because the method is SDG-specific, but because this domain exhibits the exact properties that stress-test grounded report writing: (1) *evidence intensity*: claims must be supported by heterogeneous sources (papers, official documents, tables, figures); (2) *quantitative rigor*: reports contain many numbers (indicator values, trends, comparisons) that are easy to hallucinate or mis-state; (3) *multimodal reliance*: high-quality reports heavily use figures/tables for key findings, and missing visuals is a common failure of naive generation. These characteristics directly correspond to the core problems addressed by EVIREPORT: improving factual correctness, expanding evidence coverage, and integrating visual evidence.

Benchmark construction. EviReportBench is derived from authoritative real-world reports. We collect 30 reports spanning 8 representative topics from official publications released by national governments and international organizations. All reports are obtained from official websites and curated by domain experts to ensure reliability and professional quality. We evaluate generated reports along three complementary dimensions.

Factual Accuracy (claim verification). Since the primary risk in analytical report generation is producing fluent but incorrect statements, we assess factual accuracy at the *claim level*. Following the spirit of FactScore-style evaluation, we extract independent factual claims from a generated report and verify each claim against the supporting evidence (Min et al., 2023). We report both the number of extracted claims and the correctness rate, yielding an objective measure of how faithfully the report reflects verifiable facts. This dimension is most directly impacted by evidence-grounded retrieval and evidence-aware drafting.

Factual Coverage (quiz-based completeness). Accuracy alone does not capture whether the report covers the key information expected by expert-written references. We therefore design a quiz-based protocol to measure coverage with respect to human reports. For each topic, we construct 40 quizzes using predefined templates (Appendix A.6); each quiz is answered by an LLM using *only* the generated report as context. Coverage is measured by the *answerable ratio* R_a (whether the report contains sufficient information to answer) and the *accuracy* A (whether the produced answer is correct), which are combined into an overall score:

$$S = 0.25 \times R_a + 0.75 \times A. \quad (1)$$

This dimension is designed to reflect whether the outline and subsection plans lead to comprehensive, evidence-supported content rather than a generic template.

Image Recall (visual evidence integration). To quantify whether the system retrieves and integrates informative visuals, we select the highest-quality expert report per topic and extract its key figures as ground-truth visual evidence. We then extract images from generated reports and compute image recall as the proportion of ground-truth images covered by the generated output. This metric directly targets the "missing visuals" failure mode and evaluates caption-guided multimodal retrieval and integration.

4 Experiments

4.1 Experimental Settings

Implementation. In the retrieval stage, we employ `bge-large-en-v1.5` (Xiao et al., 2023) to encode queries and evidence units for semantic retrieval. For report generation, we use `DeepSeek-V3` (DeepSeek-AI, 2024) and `GPT-5` (OpenAI, 2025a) as LLM agents in the planning and writing stages. During evaluation, we adopt `Qwen-Max` (Alibaba Cloud, 2025) with web search enabled as the verification model for claim checking and evidence validation.

Baselines. We compare EVIREPORT with the following representative baselines:

Naive_RAG: Uses the same reference corpus as EVIREPORT and directly prompts an LLM to generate a full report with retrieved passages.

ReAct_RAG: A tool-using agent that iteratively calls text/image retrieval tools in a ReAct-style reasoning loop (Yao et al., 2023).

Methods	Factual Accuracy		Factual Coverage			Image Recall
	Fact Num.	Accuracy	Ans. Ratio	Accuracy	Overall Score	
Naive_RAG	22.00	0.8890	0.1713	0.0632	0.0917	0.38
ReAct_RAG	10.38	0.8311	0.3395	0.1314	0.1834	0.45
Coze	32.75	0.3466	0.3162	0.0649	0.1277	–
OpenAI_DR	53.62	0.8850	0.2274	0.0719	0.1107	–
Gemini_DR	29.00	0.6854	0.1684	0.0316	0.0658	–
EVIREPORT	**78.50**	**0.9687**	**0.6100**	**0.3250**	**0.3963**	**0.79**

Table 1: Main results on EviReportBench. We report factual accuracy, factual coverage, and image recall. "Fact Num." denotes the number of factual statements that are verified as correct in the generated report. "Ans. Ratio" denotes the proportion of questions that can be answered based solely on the generated report.

OpenAI_DR: Generates reports using `ChatGPT` Deep Research (OpenAI, 2025b).

Gemini_DR: Generates reports using `Gemini` Deep Research (Google AI for Developers, 2025).

Coze: Generates reports using the `Coze` agent platform (Coze, 2025).

Test Cases. We evaluate all methods on the 8 benchmark topics described in Sec. 3.5. For each topic, we generate a complete long-form report and evaluate it using the proposed metrics on factual accuracy, factual coverage, and visual evidence integration.

4.2 Main Results

Table 1 summarizes the main results on EviReportBench, evaluating all methods from three complementary perspectives: factual accuracy, factual coverage, and image recall.

Overall comparison. EVIREPORT consistently outperforms all baselines across all metrics by a large margin. Notably, EVIREPORT achieves both *high factual density* and *high correctness*, indicating that it improves faithfulness without sacrificing informativeness. We additionally test different backbone LLMs for generation, and the trends remain consistent; detailed results are provided in the Appendix. A.5.2.

Factual accuracy. EVIREPORT obtains the highest number of verified factual statements (**78.5**) with an accuracy of **0.9687**, substantially surpassing Naive_RAG and ReAct_RAG. This suggests that organizing evidence into compact, traceable packages and enforcing evidence-aware drafting effectively reduces unsupported statements while enabling richer factual reporting.

Factual coverage. EVIREPORT also shows a clear advantage in coverage, achieving an answerable ratio of **0.61** and an accuracy of **0.325**, resulting in the highest overall coverage score of **0.3963**. Compared with the strongest baseline ReAct_RAG, EVIREPORT improves the overall coverage score by nearly a factor of two, indicating that hierarchical planning helps allocate space to key aspects and reduces "template-like" omissions.

Deep research baselines. OpenAI_DR and Gemini_DR achieve relatively high factual accuracy but low factual coverage on our benchmark. This suggests that general-purpose deep research systems can produce locally correct statements, yet may under-cover topic-critical facts when the task requires systematic synthesis over dense quantitative and multimodal evidence. Coze and Naive_RAG show moderate performance but suffer from either low factual accuracy or limited coverage, highlighting the limitations of direct long-form generation without structured planning and evidence packaging.

Image recall. EVIREPORT achieves the highest image recall (**0.79**), outperforming Naive_RAG and ReAct_RAG by a clear margin. This demonstrates the effectiveness of caption-guided multimodal retrieval for integrating informative visual evidence into reports. We do not report image recall for OpenAI_DR, Gemini_DR, and Coze, since these systems rely on proprietary or heterogeneous corpora and do not operate on the same controlled image database as EviReportBench.

Taken together, these results show that EVIREPORT produces reports that are more accurate, more complete, and better supported by visual evidence, validating the effectiveness of the proposed retrieve–plan–write framework.

4.3 Ablation Studies

Table 2 reports ablation results on EviReportBench, quantifying the contribution of each key component in EVIREPORT to factual accuracy, factual coverage, and visual evidence integration.

Ablation Object	Factual Accuracy		Factual Coverage			Image Recall
	Fact Num.	Accuracy	Ans. Ratio	Accuracy	Overall Score	
w/o KG	67.50	0.9053	0.3221	0.1326	0.1800	0.45
w/o Fact Extraction	52.75	0.8350	0.4574	0.1419	0.2208	0.63
w/o CGIR	76.88	0.9270	0.6003	0.2785	0.3542	0.58
EVIREPORT	**78.50**	**0.9687**	**0.6100**	**0.3250**	**0.3963**	**0.79**

Table 2: Ablation study on EviReportBench. KG denotes the evidence-oriented knowledge graph used for graph-enhanced retrieval. CGIR denotes Caption-Guided Image Retrieval.

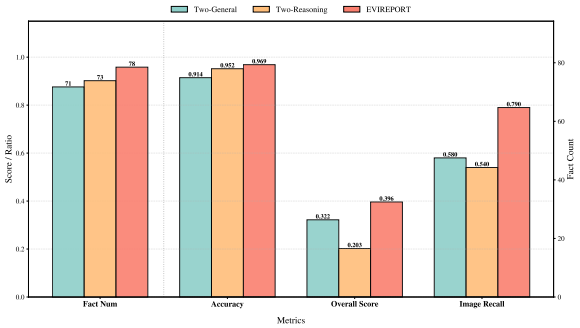


Figure 3: Ablation on two-stage outlining. "Two-General" and "Two-Reasoning" denote single-model planning variants, while EVIREPORT uses coarse-to-fine outlining.

Effect of graph-enhanced retrieval (KG). Removing the knowledge graph in the pre-planning retrieval stage (w/o KG) leads to a pronounced drop in factual coverage: the overall coverage score decreases from 0.3963 to 0.1800 and the answerable ratio drops from 0.6100 to 0.3221. This suggests that bypassing structured evidence neighborhoods weakens the system's ability to retrieve and organize sufficient, topic-relevant evidence packages. We also observe degraded multimodal integration, as image recall drops from 0.79 to 0.45, indicating that graph-guided evidence retrieval helps surface contexts that later facilitate figure discovery and selection.

Effect of two-stage outlining. Replacing our coarse-to-fine outlining with a single model degrades performance in different ways. As shown in Fig. 3, the general-only variant yields higher coverage but lower factual accuracy, while the reasoning-only variant maintains higher factual accuracy but substantially reduces coverage. This trade-off supports our design insight that global planning and fine-grained decomposition require complementary behaviors, motivating the two-stage outlining procedure.

Effect of fact extraction. Removing the explicit *fact extraction* step (Step 1) and directly drafting from retrieved context significantly hurts both reliability and completeness: factual accuracy drops from 0.9687 to 0.8350 and the number of verified facts decreases from 78.50 to 52.75, accompanied by a large reduction in coverage (overall score 0.2208 vs. 0.3963).

Effect of CGIR. Removing Caption-Guided Image Retrieval (w/o CGIR) notably reduces image recall (0.79 $\rightarrow$ 0.58) and also degrades textual quality, with factual accuracy dropping from 0.9687 to 0.9270 and the overall coverage score decreasing from 0.3963 to 0.3542. This shows that caption-guided retrieval improves figure selection and more broadly strengthens multimodal grounding for subsection drafting.

5 Conclusion and Outlook

We present EVIREPORT, an evidence-grounded workflow for automated long-form report generation that follows a retrieve–plan–write paradigm. Across all metrics on EviReportBench, EVIREPORT consistently outperforms strong baselines in factual accuracy, factual coverage, and image recall, benefiting from graph-enhanced retrieval for compact and traceable evidence packaging, coarse-to-fine outline planning, and an iterative multimodal writing loop with caption-guided image retrieval. While EviReportBench is instantiated on SDG-style, data- and figure-intensive reports, the proposed workflow is domain-agnostic and can be applied to other evidence-heavy reporting scenarios (e.g., policy analysis, scientific surveys, and industrial analytics). We hope this work provides a practical path toward more trustworthy report-generation agents and offers an evaluation blueprint for studying faithfulness, completeness, and multimodal evidence integration in long-form generation.

Limitations

This work has several limitations. First, the system does not incorporate continuously updated external data sources, limiting access to diverse and up-to-date datasets referenced in human-written SDGs reports. As a result, it cannot always compute the latest indicator values, reducing the timeliness of generated reports. Second, the generation process relies on large language models, which may still produce hallucinated or imprecise statements despite evidence-guided retrieval, affecting factual reliability in complex analytical scenarios. Future work will explore integrating a multi-agent SDGs data analysis platform with the report generation assistant to enable coordinated data acquisition, indicator computation, and verification, thereby improving the depth, timeliness, and accuracy of generated reports.

Ethical considerations

This work studies an automated workflow for generating SDGs domain reports to assist users in efficiently organizing and summarizing evidence from large collections of public sustainability documents. Our framework relies on publicly available sources, including official reports released by international organizations and governments, as well as published research papers, and we only use these materials for retrieval and grounded synthesis to respect copyright and intellectual property constraints.

The system is intended to augment, not replace, human expertise. Users should critically review generated reports, especially quantitative claims and policy recommendations, and verify them against authoritative sources before downstream use. As SDGs-related materials and indicators may contain reporting biases, temporal gaps, or regional unevenness, the generated outputs can inherit such biases or present incomplete coverage. We mitigate these risks by emphasizing evidence-grounded generation, providing traceable retrieval context, and evaluating factuality and coverage; nevertheless, we do not claim that outputs are error-free. We encourage responsible use, transparent disclosure when AI assistance is involved, and appropriate human oversight for high-stakes decision making.

References

Alibaba Cloud. 2025. Alibaba cloud model studio: Qwen-max model. https://www.alibabacloud.com/product/modelstudio. Accessed: 2025-12-20.

CBAS. International research center of big data for sustainable development goals (cbas). https://www.cbas.ac.cn/en/. Accessed: 2025-12-22.

Coze. 2025. Coze space: Ai agent platform. https://space.coze.cn/. Accessed: 2025-12-20.

DeepSeek-AI. 2024. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2024. Chain-of-verification reduces hallucination in large language models. In *Findings of the association for computational linguistics: ACL 2024*, pages 3563–3578.

Zican Dong, Tianyi Tang, Junyi Li, Wayne Xin Zhao, and Ji-Rong Wen. 2024. Bamboo: A comprehensive benchmark for evaluating long text modeling capacities of large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2086–2099.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2024. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pages 6491–6501.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2023a. Rarr: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508.

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

Zengyi Gao, Yukun Cao, Hairu Wang, Ao Ke, Yuan Feng, S. Kevin Zhou, and Xike Xie. 2025. Frag: A flexible modular framework for retrieval-augmented generation based on knowledge graphs. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 6178–6192.

Google AI for Developers. 2025. Gemini deep research agent. https://ai.google.dev/gemini-api/docs/deep-research. Accessed: 2025-12-20.

Yuntong Hu, Zhihan Lei, Zheng Zhang, Bo Pan, Chen Ling, and Liang Zhao. 2024. Grag: Graph retrieval-augmented generation. *arXiv preprint arXiv:2405.16506*.

Haibo Jin, Haoxuan Che, Yi Lin, and Hao Chen. 2024. Promptmrg: Diagnosis-driven prompts for medical report generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 2607–2615.

Ishita Kumar, Snigdha Viswanathan, Sushrita Yerra, Alireza Salemi, Ryan A Rossi, Franck Dernoncourt, Hanieh Deilamsalehy, Xiang Chen, Ruiyi Zhang, Shubham Agarwal, and 1 others. 2024. Longlamp: A benchmark for personalized long-form text generation. *arXiv preprint arXiv:2407.11016*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33.

Mufei Li, Siqi Miao, and Pan Li. 2025. Simple is effective: The roles of graphs and large language models in knowledge-graph-based retrieval-augmented generation. In *International Conference on Learning Representations (ICLR)*.

Lei Liang, Mengshu Sun, Zhengke Gui, Zhongshu Zhu, Zhouyu Jiang, Ling Zhong, Yuan Qu, Peilong Zhao, Zhongpu Bo, Jin Yang, and 1 others. 2024. Kag: Boosting llms in professional domains via knowledge augmented generation. *arXiv preprint arXiv:2409.13731*.

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. *arXiv preprint arXiv:2303.17651*.

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

Junbo Niu, Zheng Liu, Zhuangcheng Gu, Bin Wang, Linke Ouyang, Zhiyuan Zhao, Tao Chu, Tianyao He, Fan Wu, Qintong Zhang, Zhenjiang Jin, Guang Liang, Rui Zhang, Wenzheng Zhang, Yuan Qu, Zhifei Ren, Yuefeng Sun, Yuanhong Zheng, Dongsheng Ma, and 42 others. 2025. Mineru2.5: A decoupled vision-language model for efficient high-resolution document parsing. *Preprint*, arXiv:2509.22186.

OpenAI. 2025a. Gpt-5 system card. System card, OpenAI. Canonical PDF; version dated Aug. 13, 2025 (accessed 2025-12-20).

OpenAI. 2025b. Introducing deep research. https://openai.com/index/introducing-deep-research/. Accessed: 2025-12-20.

Boci Peng, Yun Zhu, Yongchao Liu, Xiaohe Bo, Haizhou Shi, Chuntao Hong, Yan Zhang, and Siliang Tang. 2024. Graph retrieval-augmented generation: A survey. *ACM Transactions on Information Systems*.

Yijia Shao, Yucheng Jiang, Theodore Kanell, Peter Xu, Omar Khattab, and Monica Lam. 2024. Assisting in writing wikipedia-like articles from scratch with large 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)*, pages 6252–6278.

Yushi Sun, Kai Sun, Yifan Ethan Xu, Xiao Yang, Xin Luna Dong, Nan Tang, and Lei Chen. 2025. Kerag: Knowledge-enhanced retrieval-augmented generation for advanced question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 6194–6216.

Bowen Tan, Zichao Yang, Maruan Al-Shedivat, Eric Xing, and Zhiting Hu. 2021. Progressive generation of long text with pretrained language models. In *Proceedings of the 2021 conference of the North American chapter of the association for computational linguistics: human language technologies*, pages 4313–4324.

Chen Tang, Chenghua Lin, Henglin Huang, Frank Guerin, and Zhihao Zhang. 2022. Etrica: Event-triggered context-aware story generation augmented by cross attention. *arXiv preprint arXiv:2210.12463*.

United Nations Department of Economic and Social Affairs. 2025. The sustainable development goals report 2025. Released 14 July 2025.

United Nations General Assembly. 2017. Work of the statistical commission pertaining to the 2030 agenda for sustainable development (a/res/71/313). https://digitallibrary.un.org/record/1291226/files/A_RES_71_313-EN.pdf. Accessed: 2025-12-20.

United Nations Secretary-General. 2025. Progress towards the sustainable development goals: Report of the secretary-general. 29 April 2025.

United Nations Statistics Division. Sdg indicators: Global indicator framework and indicator list. https://unstats.un.org/sdgs/indicators/indicators-list/. Accessed: 2025-12-20.

Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Qingsong Wen, Wei Ye, and 1 others. 2024. Autosurvey: Large language models can automatically write surveys. *Advances in neural information processing systems*, 37:115119–115145.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *Preprint*, arXiv:2309.07597.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R. Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*.

Ruochen Zhao, Xingxuan Li, Shafiq Joty, Chengwei Qin, and Lidong Bing. 2023. Verify-and-edit: A knowledge-enhanced chain-of-thought framework. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5823–5840.

A Appendix

Due to space limitations in the main paper, we provide additional details in the appendix:

- Sec. A.1 Algorithms (pseudocode).
- Sec. A.2 Database Construction.
- Sec. A.3 SDGs Domain Knowledge Graph Construction for Report Generation.
- Sec. A.4 Details of Evaluation Data.
- Sec. A.5 Additional Results (hyperparameters, model variants, per-topic analysis).
- Sec. A.6 Quiz Construction for Factual Coverage Evaluation.
- Sec. A.7 Prompts Used.

A.1 Algorithms

This appendix provides implementation-level pseudocode for the key modules of EVIREPORT, complementing the high-level descriptions in Sec. 3.2–Sec. 3.4. Algorithm 1 details graph-enhanced evidence retrieval that produces compact, traceable evidence packages for planning and writing. Algorithm 2 specifies the subsection drafting procedure with the three-step iterative loop and caption-guided multimodal retrieval.

Algorithm 1 Graph-Enhanced RAG Retrieval (KG-RAG)

Input: query q, knowledge graph $\mathcal{G}$, text database $\mathcal{D}$, top-K nodes K, hop size h
Output: evidence pairs $\mathcal{P} = \{(\mathcal{G}_i, s_i)\}$
Extract keywords $\mathcal{K}$ from q and compute query embedding $\mathbf{e}_q$
Retrieve candidate *case* nodes from $\mathcal{G}$ using $\mathcal{K}$ and $\mathbf{e}_q$
Merge and deduplicate retrieved nodes to obtain $\mathcal{E}$; keep top-K by relevance
$\mathcal{P} \leftarrow \emptyset$
for each case node $c \in \mathcal{E}$ **do**
    Extract an entity-centered subgraph $\mathcal{G}_c$ from $\mathcal{G}$ within h hops of c
    Prune $\mathcal{G}_c$ by removing low-relevance nodes/edges w.r.t. q to obtain $\mathcal{G}'_c$
    Use $\mathcal{G}'_c$ to form retrieval constraints (entities/relations) and retrieve text blocks $\mathcal{B}_c$ from $\mathcal{D}$
    Summarize $\mathcal{B}_c$ into s_c (key facts, numbers, and missing information) conditioned on q
    Augment $\mathcal{G}'_c$ with summary-derived attributes to obtain $\mathcal{G}''_c$
    $\mathcal{P} \leftarrow \mathcal{P} \cup \{(\mathcal{G}''_c, s_c)\}$
end for
Return: $\mathcal{P}$

A.2 Database Construction

To ensure the quality and professionalism of AI-generated reports, we build both a text retrieval database and an image retrieval database.

Data collection. We collect (i) 109,731 SDGs-related research papers, (ii) 14 sustainability reports (2019–2025) released by the International Research Center of Big Data for Sustainable Development Goals (CBAS)(CBAS), including China and international SDGs reports, (iii) three sustainability reports (2020–2022) for the "Belt and Road" initiative, and (iv) 687 documents published by the United Nations, other international organizations, and national governments.

Document parsing and text indexing. The collected documents exhibit diverse and complex formats. After preliminary investigation, we adopt a hybrid pipeline combining MinerU2.5(Niu et al., 2025) and a vision-language model (VLM) to convert all sources into unified Markdown (MD) files. Specifically, we first apply MinerU2.5 to extract text, tables, and figures, and then verify the con-

Algorithm 2 Content generation with a three-step loop

Input: outline $\mathcal{O} = \{o_1, \ldots, o_N\}$; text DB $\mathcal{D}_t$; image DB $\mathcal{D}_v$; max iterations T
Output: final report $\mathcal{R}$
Initialize generated subsection contents
$\mathcal{C} \leftarrow \emptyset$
for $i = 1$ to N **do**
  Construct retrieval queries q_i^{text} and q_i^{img} from $(o_i, \mathcal{C}_{<i})$
  Retrieve text blocks
  $\mathcal{B}_i \leftarrow \textsc{Search}(\mathcal{D}_t, q_i^{\text{text}})$
  Retrieve images
  $\mathcal{I}_i^{(0)} \leftarrow \textsc{Search}(\mathcal{D}_v, q_i^{\text{img}})$
  Extract caption-like strings from text
  $\mathcal{C}_i^{\text{cap}} \leftarrow \textsc{ExtractCaptions}(\mathcal{B}_i)$
  Caption-based image retrieval
  $\mathcal{I}_i^{(1)} \leftarrow \textsc{Search}(\mathcal{D}_v, \mathcal{C}_i^{\text{cap}})$
  $\mathcal{I}_i \leftarrow \textsc{Dedup}(\mathcal{I}_i^{(0)} \cup \mathcal{I}_i^{(1)})$
  $\tilde{\mathcal{B}}_i \leftarrow \textsc{Clean}(\mathcal{B}_i)$
  $\tilde{\mathcal{I}}_i \leftarrow \textsc{PackMeta}(\mathcal{I}_i)$
  Assemble context $\mathcal{X}_i \leftarrow [\tilde{\mathcal{B}}_i; \tilde{\mathcal{I}}_i; \mathcal{C}_{<i}]$
  $d_i \leftarrow \emptyset$; $m_i \leftarrow \emptyset$
  for $t = 1$ to T **do**
    Step 1 (Facts):
    $F_i \leftarrow \text{LLM\_}\textsc{Facts}(\mathcal{X}_i, o_i)$
    Step 2 (Draft):
    $d_i \leftarrow \text{LLM\_}\textsc{Write}(F_i, \mathcal{X}_i, o_i)$
    Step 3 (Append):
    $(m_i, q_i^{\text{app}}) \leftarrow \text{LLM\_}\textsc{AppendQuery}(d_i, F_i, o_i)$
    if $m_i = \emptyset$ **then**
      break
    end if
    Retrieve additional evidence
    $\Delta\mathcal{B}_i \leftarrow \textsc{Search}(\mathcal{D}_t, q_i^{\text{app}})$
    $\mathcal{X}_i \leftarrow [\mathcal{X}_i; \textsc{Clean}(\Delta\mathcal{B}_i)]$
  end for
  $c_i \leftarrow \textsc{Finalize}(d_i, \mathcal{X}_i)$
  $\mathcal{C} \leftarrow \mathcal{C} \cup \{c_i\}$
end for
$\mathcal{R} \leftarrow \textsc{Refine}(\mathcal{C})$
Return: $\mathcal{R}$

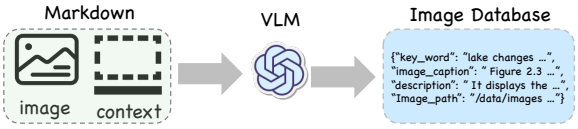


Figure 4: Pipeline for image database construction. We extract each image together with its caption and surrounding context from the parsed Markdown documents, and use a vision–language model (VLM) to generate keywords and a semantic summary for indexing.

verted content, focusing on error-prone elements such as tables and images. When conversion errors are detected, we invoke the VLM for format correction. This hybrid design balances efficiency and accuracy: the VLM provides higher fidelity but is constrained by token budget and latency, so we only use it for difficult cases.

We then split each MD document into chapter-aware chunks, resulting in approximately two million text blocks. For each block, we prepend the document title and its ancestor section titles, preserving hierarchical structure for downstream retrieval. Next, we encode the text blocks using the `bge-large-en-v1.5` embedding model and index them in Milvus for dense retrieval. In addition, we index the raw text in Elasticsearch to enable hybrid sparse–dense retrieval.

Image database construction. As illustrated in Fig. 4, we extract each image together with its caption and surrounding context paragraphs from the MD files. We feed the image and its textual context into a VLM to generate a structured image summary and keywords. Finally, we store the image caption, VLM-generated summary, and image path in an image database to support caption-guided image retrieval.

A.3 Knowledge Graph Construction

This section details how we construct the SDGs domain knowledge graph (SDG-KG) used in EVIREPORT for retrieval and evidence organization during report generation. SDG-KG is designed to explicitly model the evidence chain behind SDGs reporting, i.e., linking official indicators to real-world cases and further grounding each case with datasets and analytical methods.

A.3.1 Base SDGs schema from the official framework

We start from the official SDGs global indicator framework, which provides a stable hierarchical

mapping from *Goals* to *Indicators*(United Nations Statistics Division; United Nations General Assembly, 2017). We parse the framework documents and build a canonical hierarchy consisting of:

- **Goal** nodes (17 Goals), each associated with goal titles and descriptions;
- **Indicator** nodes, associated with indicator codes (e.g., `11.3.1`), indicator names, and textual definitions;
- **Goal–Indicator** edges representing the official hierarchy.

This base schema forms the backbone of SDG-KG and serves as an anchor for integrating evidence extracted from external references.

A.3.2 Evidence-oriented extension: Indicator–Case–Dataset–Method

Human-written SDGs analyses and policy reports typically discuss indicator progress through concrete *cases* (e.g., a country, city, region, or project), supported by specific *datasets* and *methods* (e.g., statistical models, remote sensing pipelines, machine learning algorithms). To reflect this writing pattern, we extend the base SDG hierarchy into an evidence-oriented knowledge structure with four key entity types:

- **Case**: a concrete analysis scenario (e.g., country/region/city, sector, or program) used to assess one or more indicators;
- **Dataset**: the data source(s) used to measure or estimate the indicator(s) for a given case;
- **Method**: the analytical or computational approach applied to the dataset(s) (e.g., regression, index construction, causal estimation, satellite-based mapping, ML models);
- **Indicator**: the official SDG indicator being assessed.

We then model their relationships as:

- **RELATED_TO** edges between **Indicator** and **Case**, capturing that a case study reports or evaluates the indicator;
- **USE_DATASET** edges between **Case** and **Dataset**;
- **USE_METHOD** edges between **Case** and **Method**.

As a result, each indicator is associated with a collection of evidence subgraphs, enabling the system to retrieve not only definition-level information, but also grounded analytical evidence used in real SDGs reporting.

A.3.3 Entity and relation extraction from references

To populate SDG-KG beyond the official framework, we automatically extract entities and relations from the collected SDGs-related papers and reports. Specifically, given a parsed document in Markdown, we perform:

1. **Candidate entity mining.** We use an LLM-based extractor to identify mentions of cases, datasets, and methods in paragraphs that discuss SDGs indicators. The extractor produces typed candidates (Case/Dataset/Method) together with supporting spans.
2. **Indicator linking.** We link extracted evidence to indicator nodes by matching indicator codes/names and by semantic similarity between the evidence context and indicator descriptions. When multiple indicators are plausible, we keep top candidates and resolve them in the fusion step (Sec. A.3.4).
3. **Relation construction.** We create **Indicator–Case**, **Case–Dataset**, and **Case–Method** edges when the corresponding entities co-occur within a coherent local context (e.g., within the same section or a bounded window of paragraphs), and the LLM extractor provides a positive relation decision with minimal justification.

A.3.4 Entity fusion, deduplication, and incremental updates

Since extracted entities may contain aliases and surface-form variations (e.g., "LCRPG" vs. "Land Consumption Rate to Population Growth Rate"), we perform entity fusion and deduplication before insertion. We combine:

- **String-level normalization** (case folding, punctuation removal, acronym expansion when available);
- **Embedding-based similarity** to detect near-duplicate entities;
- **Type-aware constraints** to prevent merging across incompatible types (e.g., Method vs. Dataset).

After fusion, we incrementally insert new nodes/edges into SDG-KG while preserving the canonical Goal–Indicator backbone.

A.3.5 Example: an evidence subgraph

Fig 5 illustrates an example evidence subgraph in SDG-KG (English version). In this example, the **Goal** node `SDG 11.3` is linked to its official **Indicator** node `SDG 11.3.1` via the HAS_INDICATOR relation. We further connect a concrete **Case** node (e.g., "urban land-use efficiency in China") to both the goal and the indicator using RELATED_TO, reflecting that a case study is typically written to assess progress under a specific goal/indicator.

Crucially, the case is grounded by explicit **Dataset** and **Method** nodes. As shown, the case is associated with multiple datasets (e.g., land-cover products, national geographic conditions monitoring data, and statistical yearbooks) through USE_DATASET. Meanwhile, the analytical process is represented by method nodes (e.g., multi-scale semantic segmentation, machine-learning algorithms, and built-up area extraction pipelines) connected via USE_METHOD. In addition, metric definitions used for indicator computation (e.g., LCRPGR and BPC metrics) are also modeled as method/measurement nodes and linked to the case. This evidence-oriented structure enables SDG-KG to retrieve not only indicator definitions, but also representative cases and their supporting datasets/methods, thereby improving evidence coverage for downstream report generation.

A.4 Details of Evaluation Data

We select eight representative SDGs report cases from authoritative sustainability reports and related technical assessments, as summarized in Table 3. These cases span diverse SDG themes (e.g., food production, water environment, building electrification, urban development, climate extremes) and cover multiple evidence types, including indicator definitions, datasets, analytical methods, quantitative results, and key figures (maps, trend curves, and spatial comparisons). Such diversity enables a comprehensive evaluation of long-form report generation in terms of *factual accuracy*, *factual coverage*, and *multimodal evidence integration*.

How the cases are used. In our experiments, each case topic is used as an input query for report generation, ensuring that all compared methods produce reports under the same thematic requirements. The corresponding human-written reference reports serve two purposes: (i) they provide source material for constructing the quiz-based factual coverage benchmark (Sec. A.6); and (ii) for image recall evaluation, we extract key figures from the highest-quality reference report for each topic as ground-truth visual evidence, and then compute the fraction covered by figures included in generated reports.

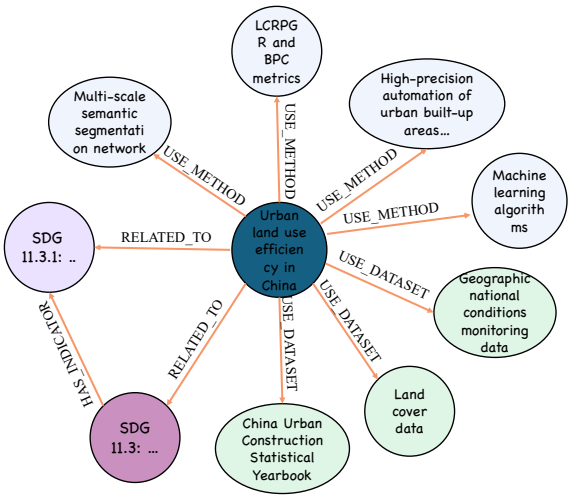


Figure 5: An example evidence subgraph in SDG-KG. The goal `SDG 11.3` links to its indicator `SDG 11.3.1` via HAS_INDICATOR. A case node is connected to the goal/indicator via RELATED_TO, and further grounded by dataset nodes (USE_DATASET) and method/measurement nodes (USE_METHOD).

A.5 Additional Results

A.5.1 Iteration Budget for the Three-step Loop

We study the iteration budget T of the three-step writing loop (facts $\rightarrow$ draft $\rightarrow$ append query) by varying the number of loop iterations from $T{=}1$ to $T{=}5$ while keeping all other settings fixed (same outline, retrieval setup, and evaluation protocol). The results are summarized in Table 4.

Overall, increasing T substantially improves report quality in the low-iteration regime. Moving from $T{=}1$ to $T{=}3$ consistently increases factual density and completeness: the number of verified facts grows from 61.25 to 78.50, factual accuracy improves from 0.9316 to 0.9687, the answerable ratio rises from 0.4781 to 0.6010, and image recall increases from 0.54 to 0.79. This indicates that iterative gap-aware retrieval effectively recovers missing evidence and reduces weakly supported statements when the draft is still under-covered.

Topic	Reference report	Example figure
Goal 2 Zero Hunger	Global Farmland Change Assessment (1985–2020)	Statistical analysis of changes in total cultivated land area across six continents (1985–2020)
Goal 2 Zero Hunger	Global High-Resolution Remote Sensing Monitoring and Assessment of Multiple Crop Reclamation in Farmland (2020)	Proportion of replanted arable land in each continent relative to global replanted area
Goal 2 Zero Hunger	Identification of stable and vulnerable food production zones in Africa	Distribution patterns of maize, wheat, rice, and soybean yields in Africa
Goal 6 Clean Water and Sanitation	Spatiotemporal distribution of algal blooms in typical lakes worldwide	Annual variation rate of water transparency across regions and climate zones (2000–2021)
Goal 7 Affordable and Clean Energy	Global status of building electrification	Global unelectrified building area percentage (2020)
Goal 11 Sustainable Cities and Communities	Global nighttime light intensity changes	"NPP-VIIRS-like" nighttime light brightness distribution map
Goal 13 Climate Action	Analysis of global heat wave disaster changes and their impacts	Average annual probability of a global heat wave
Goal 2 Zero Hunger	spatiotemporal dynamics of rice cultivation patterns in the Indochina Peninsula	Spatial distribution of rice planting pattern changes in the Indochina Peninsula (2000–2019)

Table 3: Evaluation cases (topics) used for benchmarking SDGs domain report generation. Each case is associated with an authoritative reference report and representative figures used for quiz construction and image recall evaluation.

Loop Num.	Factual Accuracy		Factual Coverage			Image Recall
	Fact Num.	Accuracy	Ans. Ratio	Accuracy	Overall Score	
1	61.25	0.9316	0.4781	0.3137	0.3548	0.54
2	69.25	0.9568	0.5250	0.3214	0.3723	0.71
3	**78.50**	**0.9687**	**0.6010**	**0.3250**	**0.3963**	0.79
4	78.30	0.9462	0.5813	0.3065	0.3752	0.79
5	82.49	0.9374	0.5750	0.3207	0.3843	**0.83**

Table 4: Effect of iteration budget T for the three-step writing loop on EviReportBench.

Beyond T=3, we observe diminishing returns and a mild trade-off between metrics. While T=5 further increases the number of verified facts (82.49) and yields the best image recall (0.83), it slightly reduces factual accuracy (0.9374) and lowers the overall coverage score (0.3843) compared to T=3 (0.3963). Since each additional iteration introduces extra model calls and retrieval steps, we use T=3 as the default in the main experiments, as it provides the best balance between factual correctness, coverage, and computational cost.

A.5.2 Backbone LLM Variants

We evaluate the robustness of EVIREPORT to different backbone LLMs by replacing the generation model used in the planning and writing stages while keeping the retrieval setup, iteration budget (T=3), and evaluation protocol unchanged. Table 5 reports the results.

Overall, EVIREPORT is relatively stable across strong backbones: Gemini-2.0-flash, Claude-sonnet-4, and GPT-5.0 all achieve high factual accuracy ($\geq$0.965) and comparable factual coverage. GPT-5.0 yields the best overall balance, achieving the highest factual accuracy (0.9687) and the best coverage overall score (0.3963). Gemini-2.0-flash achieves the highest answerable ratio (0.6125) and the best image recall (0.83), suggesting stronger multimodal retrieval/selection synergy under this backbone, while its overall coverage score is slightly below GPT-5.0. Claude-sonnet-4 performs competitively but is consistently lower on coverage metrics. DeepSeek-v3.1 shows a larger

LLM	Factual Accuracy		Factual Coverage			Image Recall
	Fact Num.	Accuracy	Ans. Ratio	Accuracy	Overall Score	
Gemini-2.0-flash	77.25	0.9656	**0.6125**	0.3188	0.3922	**0.83**
Claude-sonnet-4	75.50	0.9664	0.5813	0.3063	0.3751	0.79
GPT-5.0	**78.50**	**0.9687**	0.6010	**0.3250**	**0.3963**	0.79
DeepSeek-v3.1	70.25	0.9574	0.5594	0.2813	0.3508	0.67

Table 5: Backbone LLM variants for EVIREPORT on EviReportBench (iteration budget T=3).

drop in coverage and image recall, indicating that the downstream writing loop benefits from stronger instruction-following and gap-aware querying capability. Based on these results, we use GPT-5.0 as the default backbone in the main experiments.

A.5.3 Per-topic Performance and Radar Plots

To provide a fine-grained view beyond averaged scores, we report per-topic results on the 8 benchmark topics using radar plots (Fig. 6). We visualize four complementary metrics: (a) **Fact Num.** (verified factual statements), (b) **Factual Accuracy** (claim correctness rate), (c) **Overall Score** for factual coverage (combining answerable ratio and accuracy), and (d) **Image Recall** (coverage of topic-specific ground-truth figures). Each axis corresponds to one topic, allowing direct inspection of where a method performs strongly or fails.

Overall, EVIREPORT shows consistently high performance across topics on factual accuracy and coverage-related metrics, indicating that the retrieve–plan–write workflow generalizes beyond a small subset of "easy" topics. We also observe that some baselines exhibit larger topic variance: while they may perform competitively on a few topics, they often drop sharply on others, suggesting sensitivity to evidence density and topic structure. Finally, the image-recall plot highlights that multimodal performance is highly topic-dependent and that caption-guided image retrieval improves figure coverage particularly on topics where key findings are primarily communicated through maps or trend plots.

A.6 Quiz Construction

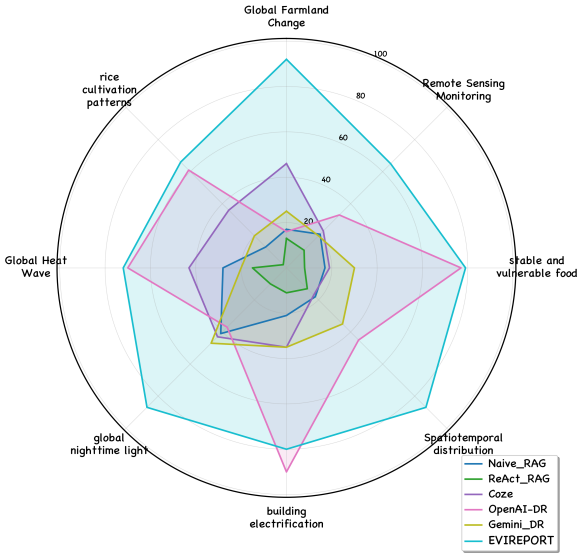
Evaluating the *factual coverage* of long-form SDGs reports is challenging because (i) gold answers are often non-unique, and (ii) free-form judging can be subjective. Following the quiz-driven, reader-aligned evaluation philosophy (i.e., a report is useful if it can answer readers' concrete questions), we construct an *evidence-constrained* quiz set and evaluate whether a generated report can answer these quizzes faithfully.

Quiz perspectives. We design six SDGs-specific quiz perspectives that target typical information needs in SDGs reporting, including indicator definitions, datasets, methods, indicator values, comparisons, and limitations. Table 6 summarizes the perspectives, descriptions, and examples.

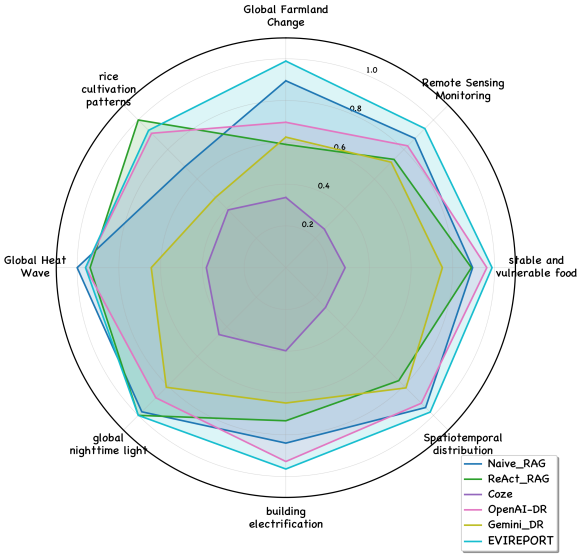
Source reports and paragraph selection. For each topic, we start from an expert-selected human-written report (officially released on authoritative websites). We segment the report into paragraphs (by section boundaries) and retain paragraphs that are information-rich and self-contained. Inspired by prior quiz construction practice, we filter out paragraphs that are too short or lack concrete signals (e.g., missing numeric values, methods, datasets, or explicit claims).

Quiz generation with evidence grounding. Given a retained paragraph p, we prompt an LLM to generate multiple quiz pairs $\{(q_j, a_j)\}_{j=1}^{k}$ under the perspective templates in Table 6, with the **strict constraint** that each answer a_j must be *fully supported* by p (no external knowledge) and should be *specific* (preferably including explicit entities, numbers, datasets, methods, time ranges, or uncertainty statements). This design reduces the ambiguity of open-ended judging and aligns the evaluation with verifiable evidence.

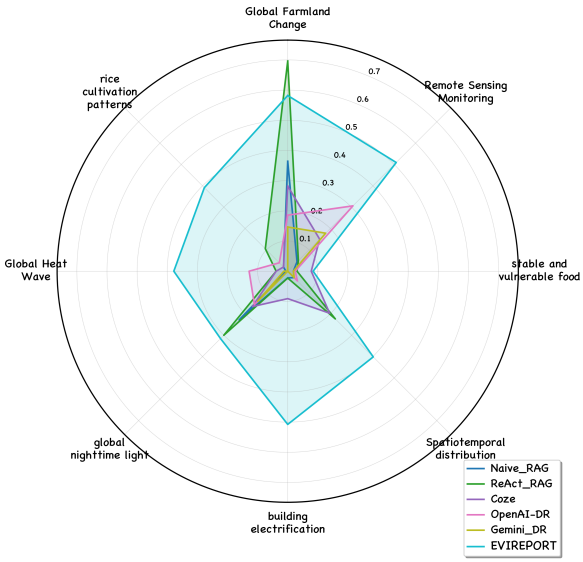
Multi-criteria validation and de-duplication. Each generated (q, a) pair is validated to ensure quality and faithfulness: (1) **format & length**: question/answer exceed minimal length and avoid vague wording; (2) **specificity**: answers contain concrete indicators (e.g., numeric values, named datasets/methods, time ranges); (3) **grounding**: keyword/phrase overlap and consistency checks between (q, a) and paragraph p; (4) **de-duplication**: remove near-duplicate quizzes within



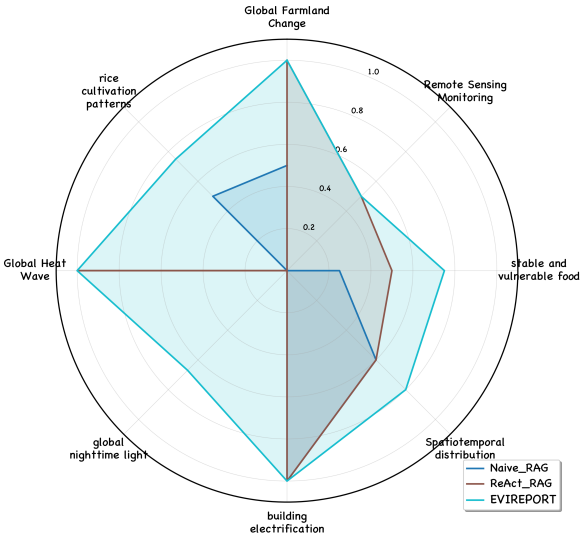
(a) Fact Num.



(b) Factual Accuracy.



(c) Overall Score.



(d) Image Recall.

Figure 6: Per-topic radar plots on EviReportBench. (a) Fact Num., (b) Factual Accuracy, (c) Overall Score, and (d) Image Recall.

a topic/perspective. We iterate generation and validation until reaching a target number of quizzes per topic (e.g., 40).

Using quizzes for factual coverage evaluation. Given a generated report $\hat{R}$, we ask an evaluation LLM to answer each quiz *using only* $\hat{R}$ (and the retrieved supporting context if applicable), and to return a fixed fallback response such as `No relevant content found in the report` when the report does not contain sufficient information. We then compute: (i) **AnswerableRatio**: the fraction of quizzes that are answerable from $\hat{R}$, and (ii) **Accuracy**: the correctness of the produced answers (optionally with evidence checks), which together reflect whether the report *covers* the key factual points expected from expert-written SDGs reports.

A.7 Prompts Used

This section summarizes the main prompts used in EviReport for (i) outline generation, (ii) content generation, and (iii) evaluation. Full prompt templates are provided after this subsection.

Outline generation. We condition outlining on retrieved evidence (KG subgraphs and summarized text blocks) and structural priors from high-quality SDGs reports. Following our two-stage strategy, a reasoning-oriented LLM first produces a high-level plan that covers essential SDGs perspectives (e.g.,

Perspective	Description	Example
Definition	Examine whether the report clearly defines SDG indicators, terminologies, measurement units, and versions used in data reporting.	What is the unit of measurement for indicator 11.3.1 on land use efficiency in China?
Dataset	Verify that dataset sources, temporal coverage, and spatial resolution are accurately stated and traceable to authoritative repositories.	Which dataset and time period are used to estimate the urban land use efficiency index?
Method	Assess whether the report describes analytical methods, models, equations, and validation metrics in sufficient detail to ensure reproducibility.	What model or formula is used to calculate the SDG indicator value?
Indicator Value	Check whether the report presents specific numerical values or rates for defined countries, years, or regions.	What was the reported land use efficiency value for Beijing in 2020?
Comparison	Evaluate whether the report analyzes temporal changes or spatial differences among SDG indicators, highlighting significant trends.	How did urban land use efficiency change between 2000 and 2020 across major cities?
Limitation	Identify acknowledgments of data gaps, uncertainties, biases, or limitations in indicator interpretation and model generalization.	What uncertainties or data limitations are mentioned in the SDG 11.3.1 analysis?

Table 6: Perspectives and examples used for evaluating factual coverage in SDGs domain reports.

indicator definition, data/methods, findings, trends, limitations, and implications). A chat-oriented LLM then refines it into a detailed hierarchical outline with non-overlapping subsection scopes and explicit writing goals, ensuring coherence and reducing redundancy.

Content generation. For each subsection, we retrieve subsection-specific text and image evidence and generate content sequentially to maintain global consistency. A CoT-style prompt guides the model to (1) extract factual points from the provided evidence, (2) draft grounded content, and (3) list missing/uncertain information. We further apply an append-and-rewrite prompt that issues a follow-up query, incorporates additional evidence, and rewrites the draft to improve completeness and factual consistency.

Evaluation. We use prompts for factual verification and quiz-based factual coverage. The verifier extracts atomic claims and checks support via evidence retrieval (with web search when enabled). For coverage, the evaluator answers each quiz using only the generated report, enabling AnswerableRatio and Accuracy computation.

Generation Prompt

Outline stage-1:

```
OUTLINE_STAGE_1_GENERATE_PROMPT= ' ' '
You are a professional assistant for generating article outlines. Based on the user's query and
the provided knowledge points, please generate an article outline.

I have constructed a knowledge graph based on existing information, which includes data sources,
computational methods used in relevant SDG case studies, and the specific SDG goals they
correspond to (e.g., SDG 11.1). I will provide you with retrieved information from the knowledge
graph as well as reference texts.

The topic is: {query}. Please help generate chapter titles and outlines accordingly. In each
chapter outline, make sure to **retain any image captions** from the provided reference
content.

Please follow the chapter structure below:
1. Target
2. Background
3. Data Used
4. Research Methods (retain relevant image captions when describing the methods)
5. Results and Analysis (retain relevant image captions in the analysis to support conclusions)
6. Key Contributions
7. Discussion and Outlook

**Knowledge Graph Retrieval Information:**
{kg_info}
' ' '
```

Outline stage-2:

```
OUTLINE_STAGE_2_GENERATE_PROMPT= ' ' '
Please generate a well-structured chapter-by-chapter writing plan in **JSON format** based on
the report topic below.
You will be provided with **entity nodes and relationship information** retrieved from a knowledge
graph, as well as **reference context** retrieved from the knowledge base.
Additionally, you may refer to the initial version of the outline generated by another LLM.
**Requirements:**

1. **Strictly output in JSON format**, avoiding extra line breaks, explanatory texts, or formatting
errors.

2. The **chapter structure should be clear**, including:

- `chapter_title`: The chapter title (must match the provided title exactly).

- `content_structure`: A structural guide for the chapter (e.g., whether it has sub-sections, brief
or detailed content, etc.).

- `content_points`: A detailed outline of the key points that must be included in this chapter; avoid
vague or generic descriptions.

3. **Chapter titles must match the provided ones exactly. Do not alter them. The structure must
include the following chapters: Target, Background, Data Used, Methodology, Results and Analysis,
Key Contributions, Discussion and Outlook.**

**Output Example:**

```json
```
```

[
{{
"chapter_title": "Target",
"content_structure": "As the first section of the article, this should generate the article title. It should correspond to the specific SDG target. No sub-sections (i.e., no ### headers); provide a brief description.",
"content_points": ...
}},
{{
"chapter_title": "Background",
"content_structure": "No sub-sections (i.e., no ### headers); provide a brief description in 1-2 paragraphs.",
"content_points": ...
...
}}
]
Please strictly follow the JSON format. Ensure `chapter_title` matches exactly with the provided titles. `content_structure` should summarize the content framework for each chapter, and `content_points` should include quantitative data or methodological details where possible—avoid vague or generic language.

Report Topic: {}

Reference Context: {}

Reference Outline (Generated by another LLM for reference):{}

' ' '

Content:

CONTENT_GENERATE_PROMPT = ' ' '
You are a professional research assistant specializing in **Retrieval-Augmented Generation (RAG)** for scientific writing.
Use ONLY the provided RAG knowledge base (annual SDG reports, peer-reviewed literature, books) to ensure **factual basis, authority, and contextual consistency**.

User Input (Query):
{query}

📌 Task Requirements
1) Write **in English**, using academic style that meets research standards.
2) No fixed length; prioritize clarity, depth, tight logic; avoid redundancy and fluff.
3) **Obey the chapter structure** described by `content_structure`; keep hierarchy concise.
4) **Exploit retrieved evidence** (data, methods, results, key findings). Prefer quantitative details (numbers, units, dates, spatial/temporal resolution, R^2/accuracy) over generic descriptions.
5) **Images**: insert only when valid absolute local paths are provided in retrieval results; embed as `![caption](ABS_PATH)`. Do not describe images without a path.
6) **Avoid generalizations**. Every claim must be backed by data, methods, or specific cases from the references.
7) If a computation has a formula, **list the formula and variables clearly** (plain text or LaTeX inline), and define symbols.
8) **Expand `content_points`** and organize strictly per `content_structure`.
9) Use the given **`chapter_title` verbatim**; keep sub-headings minimal.
10) **Formatting**:
- `# Title` only for the **first** chapter of the whole report
- `## Chapter Title` for main chapters
- `### Sub-chapter Title` optional
- Body in plain text (you may include inline equations if needed)

🔍 Evidence & Citation Policy (Hard Constraints)

- Write **only** from the "Reference Context" below. **Do not fabricate** sources, numbers, or quotes.
- **Each paragraph must contain at least one citation** in the format `[X]`, where `X` is the numbered block id.
- If a sentence relies on multiple references, append all: e.g., `[3][7][15]`.
- If no evidence supports a required point, do **not invent it**—state an **Evidence Gap Note** at the end instead.

🧱 Generation Procedure (You MUST follow)
Step 1 — Fact Extraction (bullet list):

Extract **at least 10 concrete, verifiable facts** from the Reference Context that directly support this chapter.

A **fact** refers to a *specific, evidence-based statement* that can be verified from the provided references.
Facts should represent **objective, quantitative, or methodologically grounded** information — not general statements or interpretations.

Each fact must:
- Contain **explicit data** (e.g., numerical values, proportions, metrics, R^2, accuracies, thresholds, resolutions, years, or geographic scope); or
- Describe **concrete methodologies or models** (e.g., regression, classification, remote sensing methods, LiDAR, MODIS, Landsat-based analysis); or
- Present **clearly bounded evidence** (temporal or spatial scope, dataset name, reference institution, or experimental setup).

Avoid vague claims or general interpretations such as "agriculture improved globally" or "climate change affects yield" — these are not facts.

When listing, follow this structure:
- Use **one bullet per fact**.
- Express each fact as a complete sentence.
- Include the supporting **reference IDs in brackets** (e.g., [7][15]).
- Do not infer beyond the given evidence.

Example:
- In 2020, the average cropland multiple cropping index in eastern China exceeded 180%, derived from MODIS 500 m time-series NDVI data between 2001–2020 [15][32].
- The LiDAR-based digital surface model (DSM) achieved an R^2 of 0.92 when validated against ground survey data in 2019 [7].
- Between 2000 and 2020, Vietnam's arable land area decreased by 5.8%, while agricultural productivity per hectare increased by 12% [30][35].

(This fact list serves as the factual foundation for the chapter and will be shown before the generated text.)

Step 2 — Fact-Based Writing:
Write the chapter strictly based on chapter Outline and the extracted facts.
Within each paragraph follow the micro-logic:
- **Fact:** state a referenced observation/data/method [X]
- **Interpretation:** explain significance/implications
- **Connection:** link to the chapter goal or relevant SDG target
Ensure every paragraph has ≥1 citation.

Step 3 — Self-Verification (silent check, then apply fixes):
- Remove or revise any sentence without support.
- Verify that numbers/units/time ranges match the cited sources.
- Ensure terminology is consistent with references (e.g., sensor names, resolutions, indices).

✅ Good vs Bad
Good = evidence-driven, quantitative, cited.
Bad = generic, uncited, or speculative. Follow the Good pattern.

📖 Research Plan
{plan}

📄 Already Generated Content
{already_generated}

🎯 Your Task
Execute the **current step** in the research plan.

✏️ Chapter Requirements
- **Chapter Title**: `{chapter_title}` (use exactly)
- **Chapter Content Structure Requirements**: `{content_structure}` (use as the internal outline)
- **Chapter Core Point Information** (expand these points with evidence):
{content_points}

🔍 Reference Context (authoritative, numbered)
{context_with_references}

{reference_instruction}

{image_info_text}

📦 Output Format (strict)
1) **Extracted Facts (with citations)** — a short bullet list.
2) **Chapter Text** — formatted with Markdown headers; every paragraph must include citations like [X].
3) **Evidence Gap Notes (if any)** — bullet any required points that could not be supported by the references, and specify what evidence would be needed (keep concise).

📌 Notes
1) If this is **not** the first chapter, do **not** output a document title (`#`); start at `## {chapter_title}`.
2) Adhere to `content_structure`; ensure smooth linkage with prior chapters; avoid repetition with "Already Generated Content".
3) Expand each argument using `content_points` + referenced data/methods/cases.
4) Do **not** write phrases like "as shown in the figure" unless an **actual image path** is inserted.
5) Prefer concrete values and methods over narrative statements.
6) All claims must be traceable to the numbered Reference Context; **no outside knowledge**.
'''

Evaluation Prompt

factual verification:

```
### Role
You are a meticulous fact-checking assistant. Your goal is to verify whether a given claim is factually correct by using **web search** and **primary/authoritative sources**. You must avoid hallucination: if evidence is insufficient or ambiguous, return "unsure".

### Task
Given the claim below, (1) design effective search queries, (2) retrieve evidence from the web, (3) assess whether the evidence supports or refutes the claim, and (4) output a strict JSON object following the schema.

### Claim
{fact}

### Verification Guidelines
1. **Decompose if needed**: If the claim contains multiple factual components (entities, numbers, dates, locations, causal relations), break it into atomic sub-claims mentally and verify each.
   - Output "true" only if **all** core sub-claims are supported.
   - Output "false" if **any** core sub-claim is contradicted by credible evidence.
   - Output "unsure" if evidence is missing, conflicting, outdated, or not authoritative enough.

2. **Source quality & independence**
   - Prefer: official organizations, academic venues, standards bodies, reputable news outlets, peer-reviewed papers, well-maintained official docs.
   - Avoid relying on a single source. Use **at least 2 independent sources** when possible.
   - If only low-quality sources exist, lower confidence or output "unsure".

3. **Time sensitivity**
   - If the claim is time-dependent (e.g., "latest version", "as of 2025"), ensure evidence is recent and include publication date when available.
   - If sources disagree across time (old vs new), treat that as potential conflict.

4. **Evidence handling**
   - Evidence must be directly relevant to the claim (not merely related).
   - Provide short excerpts (≤25 words) as "snippets" and include enough context to justify the verdict.
   - If you find contradictory evidence, include it in `conflicting_info` and lower confidence.

5. **Confidence calibration (0.0–1.0)**
   - 0.90–1.00: multiple high-quality independent sources directly confirm/deny, minimal ambiguity.
   - 0.70–0.89: good evidence but minor ambiguity (e.g., wording mismatch, limited sources).
   - 0.40–0.69: weak/indirect evidence or partial verification of sub-claims.
   - 0.00–0.39: little evidence, severe conflicts, or unclear claim → usually "unsure".

### Output Requirements
- Output **JSON only** (no markdown, no comments, no trailing commas).
- Do not fabricate URLs, titles, dates, or quotes.
- If you cannot find reliable evidence, return "unsure" with low confidence and explain what is missing.
- Keep `reasoning` concise (2–6 sentences). Do not include step-by-step hidden deliberation.
### Output JSON (strict)
{
  "claim": "{fact}",
  "answer": "true" | "false" | "unsure",
  "confidence": 0.0,
  "reasoning": "",
  "conflicting_info": "",
  "search_queries": [
    ""
  ],
  "supporting_evidence": [
    {
      "url": "",
      "title": "",
      "publisher": "",
      "date": "",
      "snippet": "",
      "stance": "support" | "refute"
    }
  ] }
```

quiz-based factual coverage:

You are an expert evaluator for Sustainable Development Goals (SDG) reports. Your job is to answer the user question using ONLY the provided Report Text. Do not use outside knowledge. Do not guess. If the report does not explicitly contain the answer, say so.

Question:
{QUESTION}

Report Text:
"""
{REPORT_TEXT}
"""

Instructions (strict):
1. Read the question and identify what type of information is requested:
   - Definition/term, dataset/source, method/formula, numeric value, time range, spatial scope, trend/comparison, limitation/uncertainty, policy implication.
2. Locate the answer by searching the Report Text:
   - Prefer exact matches; also consider paraphrases and abbreviations.
   - If multiple candidate passages exist, choose the one that most directly answers the question.
3. Extract the answer:
   - Keep it concise (typically 1–2 sentences or a short phrase/value).
   - Preserve numbers, units, years, and names exactly as written.
   - Do NOT rewrite the claim beyond minor trimming. No additional interpretation.
4. Provide evidence:
   - Copy the exact supporting sentence(s) verbatim from the Report Text.
   - If one sentence is insufficient, include up to 3 sentences, concatenated with a single space.
   - Evidence must be the minimal span that supports the answer.
5. Handle ambiguity:
   - If the report contains related content but does not fully answer the question, output "No relevant content found." (do not guess).
   - If the question asks for something not present (e.g., specific country/year not mentioned), output "No relevant content found."
6. Confidence scoring (0.0–1.0):
   - 0.90–1.00: direct explicit statement matching the question exactly.
   - 0.70–0.89: clear but slightly indirect (e.g., paraphrase, minor inference from a stated table/value).
   - 0.40–0.69: partially related; answer is incomplete → usually should return "No relevant content found."
   - 0.00–0.39: not found or highly ambiguous → return "No relevant content found."
   - If you output "No relevant content found.", do not output confidence.

Output JSON (strict; JSON only, no markdown, no extra keys):
Option A (answer found):
```
{
  "answer": "...",
  "supporting_sentence": "...",
  "confidence": 0.0
}
```

Option B (not found):
```
{
  "answer": "No relevant content found."
}
```