

DocFollow: A Hierarchical Fine-Grained Constraints Following Benchmark in Long Contexts for LLMs

Yanji He^{*1}, Zhao Yang^{*1}, Yuxin Jiang², Yiqian Yang¹, Jiaheng Wei^{†1}, Wei Wang^{‡1,3}

¹The Hong Kong University of Science and Technology (Guangzhou)

²Huawei Technologies Co.,Ltd, ³The Hong Kong University of Science and Technology

yhe720@connect.hkust-gz.edu.cn, im.yang.zhao.edu@gmail.com,

jiang.yuxin2@huawei.com, yyang937@connect.hkust-gz.edu.cn,

jiahengwei@hkust-gz.edu.cn, weiwcs@ust.hk

Abstract

Large Language Models (LLMs) have demonstrated remarkable capabilities in processing long contexts and generating long-form texts. However, existing benchmarks evaluate these capabilities by focusing on either long-input comprehension with short outputs or short-input creative generation with lengthy outputs. This critical gap overlooks real-world *Long-In-Long-Out Generation* tasks, where models must simultaneously process long-context inputs and produce long-form outputs while adhering to fine-grained constraints. We introduce *DocFollow*, a comprehensive benchmark featuring 2,436 expert-validated instances across 5 major domains and 20 subdomains, designed to evaluate LLMs’ ability to follow hierarchical constraints in *Document Following* tasks. Our benchmark employs five types of constraints with increasing granularity. To ensure reliable evaluation, we develop a rule-based assessment protocol that decomposes overall evaluation uncertainty into atomic-level classification uncertainties, employing LLMs solely as information extractors for well-established NER and RE tasks rather than subjective scoring. Our evaluation of 13 state-of-the-art LLMs reveals a consistent pattern: while models maintain reasonable performance on coarse-grained constraints, they exhibit substantial degradation on fine-grained requirements, particularly entity-level constraints. This finding suggests fundamental architectural limitations in maintaining detailed structural and semantic fidelity throughout extended generation. Our findings highlight the critical need for advancing long-context generation capabilities beyond surface-level coherence to achieve precise fine-grained control. The implementation is publicly available.[‡]

^{*}These two authors contributed equally to this work.

[†]Corresponding author.

[‡]Source code is available at <https://github.com/leonbig/DocFollow>.

Benchmark	Long I-O	Diverse Scenarios	Reliable Evaluation	Fine-Grained Constraints
FollowBench (Jiang et al., 2024)	✗	✓	✗	✗
HelloBench (Que et al., 2024)	✗	✓	✗	✗
ProxyQA (Tan et al., 2024)	✗	✗	✓	✗
LongWriter (Bai et al., 2025)	✗	✓	✓	✗
LongGenBench (Wu et al., 2025)	✗	✗	✓	✓
WritingBench (Wu et al., 2026)	✓	✓	✗	✗
<i>DocFollow (ours)</i>	✓	✓	✓	✓

Table 1: Comparison of *DocFollow* with representative text generation benchmarks. “Long I-O” refers to tasks where both average input and output exceed 1000 words. “Diverse Scenarios” means the benchmark covers diverse domains and writing scenarios. “Reliable Evaluation” denotes reliable evaluation methods that avoid direct scoring by LLMs. “Fine-Grained Constraints” refers to whether the benchmark adopts entity-level constraints.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities in processing long contexts and generating extended texts, excelling in two primary scenarios: 1) generating concise outputs from extensive inputs like summarization, and 2) generating long-form texts from minimal input like creative writing. State-of-the-art LLMs such as ChatGPT-5.1, Claude-4.5-Sonnet and DeepSeek-R1 (DeepSeek-AI, 2025) feature increasingly expansive context windows, with numerous benchmarks and studies examining their performance on both distinct scenarios. However, the intersection of these capabilities—tasks requiring both long-context input processing and long-form text generation (where both input and output exceed 1,000 words) remains critically underexplored. In this work, we refer to such tasks as *Long-In-Long-Out Generation*.

Long-In-Long-Out Generation aligns closely with real-world writing scenarios, such as drafting academic research papers that synthesize extensive documents while maintaining theoretical coherence and writing news reports that integrate

interviews and records while adhering to the narrative’s chronological sequence. The extended inputs necessitate effective retrieval and comprehension of lengthy sequences (akin to “Needle in a Haystack” (*NIAH*) (Wang et al., 2025)), while generating long outputs requires precise instruction following, particularly for fine-grained constraints (analogous to “Reversed *NIAH*”²). Despite the practical importance of these applications, existing benchmarks predominantly focus on isolated capabilities, with most emphasizing long-input processing yielding short outputs. LLMs’ performance on integrated *Long-In-Long-Out Generation* tasks remains uncertain. The limitations of current benchmarks highlight difficulties of developing benchmarks for *Long-In-Long-Out Generation*. The first is the lack of authentic, fine-grained task design. For instance, LongGenBench examines instruction-following but restricts inputs to short texts and employs only four manually crafted query types, which fails to capture the complexity and diversity of real-world writing scenarios. The second is the absence of reliable evaluation frameworks. Critically, benchmarks like WritingBench exemplify both issues: it evaluates only coarse-grained constraints such as personalization and style, and compounds this weakness by relying entirely on unreliable LLM-as-judge methods. These approaches are inconsistent (Gu et al., 2024), while alternatives are equally problematic: expert manual review is prohibitively costly, and traditional metrics like BLEU (Papineni et al., 2002) are ill-suited for capturing semantic nuance in long-form generation.

These considerations highlight two fundamental challenges in developing *Long-In-Long-Out Generation* benchmarks:

- **[P1]** Designing tasks with *controllable, fine-grained* instructions that comprehensively assess capabilities across *diverse* text generation scenarios.
- **[P2]** Implementing *automated* evaluation methods that deliver *reliable* measurements of instruction following performance.

To address P1, as illustrated in Figure 1, we define the task of our benchmark as *Document-Following* task. In this task, models are instructed

²In the *NIAH* task, the model’s goal is to find an existing piece of information (the needle) in a long text (the haystack). In the reversed *NIAH* task, the model’s goal is to create a long text (the haystack) while correctly placing a specific piece of information (the needle) according to an instruction.

to generate new long-form text on a modified topic while adhering to multi-level constraints derived from reference documents sourced from diverse *real-world* topic domains and writing scenarios. These challenging constraints range from document-level structure (e.g., length) to entity-level precision (e.g., preserving entity relations types) and are both natural and controllable. As such, the task transcends mere summarization or creative writing, but with hierarchical degree of different and intertwining syntactic and semantic constraints that are feasible to verify but can be extremely “hard” to generate by the LLMs. To evaluate LLMs’ capability with a *fine-grain*, the task difficulty can vary by controlling the difficulty level along two key dimensions: three Topic Modification levels (R1-R3) and four Task Difficulty Levels (L1-L4).

To address P2, we develop a rule-based evaluation protocol whose core innovation lies in decomposing overall evaluation uncertainty into atomic-level classification uncertainties. Rather than relying on LLMs to directly score constraint adherence—an approach prone to inconsistency and bias—our protocol employs LLMs solely as information extractors for well-established tasks: Named Entity Recognition (NER) and Relation Extraction (RE) (Zhang et al., 2025). All scoring is then computed through deterministic rule-based programs. This decomposition ensures that evaluation reliability depends only on mature, verifiable classification subtasks, whose accuracy we validate empirically (Section 5.1).

Formally, we introduce *DocFollow*, a novel benchmark that evaluates LLMs’ ability to generate long-form text while following hierarchical fine-grained constraints through long-context input. The benchmark comprises 2,436 manually curated instances across five major domains and twenty subdomains, providing broad coverage of diverse writing scenarios. Each instance is rigorously validated by experts to balance difficulty, avoiding trivial or insurmountable challenges and ensuring that *DocFollow* provides reliable insights into LLMs’ capabilities. Table 1 summarizes how *DocFollow* compares to related text generation benchmarks.

In summary, our major contributions include:

- To the best of our knowledge, this is the first work to discuss the challenge of fine-grained constraints following in *Long-In-Long-Out Generation* task for LLMs.

- We introduce *DocFollow*, the first comprehensive *Document-Following* benchmark featuring 2,436 manually validated instances across five primary domains and twenty sub-domains, incorporating three topic modification levels and four constraint levels to probe instruction-following and text generation capabilities comprehensively.
- We develop a rule-based evaluation protocol that provides reliable and consistent assessments of fine-grained constraints adherence in *Long-In-Long-Out Generation* tasks.
- We conduct extensive evaluation of 13 state-of-the-art LLMs, revealing that models struggle significantly with *Document-Following* tasks, especially in following entity-level constraints.

2 Related Work

2.1 Instruction Following

The emergence of instruction-tuned language models (Ouyang et al., 2022; Wei et al., 2021; Taori et al., 2023; Rafailov et al., 2023; Chiang and Lee, 2023) necessitates reliable evaluation methodologies. While LLM-as-judge approaches demonstrate improved alignment with human assessments compared to traditional metrics, they suffer from systematic biases (Zheng et al., 2023; Wang et al., 2024; Stureborg et al., 2024) and fail on complex reasoning tasks (Du et al., 2023; Madaan et al., 2023). This fundamental unreliability motivates our rule-based evaluation protocol that eschews direct LLM scoring in favor of structured information extraction.

2.2 Long-text Benchmarks

Current long-context benchmarks, evolving from *NIAH* to comprehensive Long-In-Short-Out like (Bai et al., 2024; An et al., 2024; Zhang et al., 2024), primarily assess input comprehension. This leaves the critical Long-In-Long-Out setting, which mirrors real-world tasks, underexplored. The few attempts in this area are limited—for instance, by short inputs (LongGenBench) or by coarse constraints and unreliable LLM-based evaluation (WritingBench). Consequently, this paper fills a clear research gap by introducing a benchmark for fine-grained instruction following in *Long-In-Long-Out Generation* tasks.

3 The *DocFollow* Dataset

3.1 Task Formulation

The goal of our benchmark is to evaluate how well LLMs perform the *Document-Following* task. The formal definition of *Document-Following* task is: Let X denote an original document associated with an original topic T , and T' denote a modified topic derived from T . Given X , T' , and a set of constraints $\{c_1, c_2, \dots, c_n\}$ which are related to X and T' , the task requires the LLM to generate a new long document X' that satisfies all specified constraints. Formally, the input comprises the triplet $(X, T', \{c_1, \dots, c_n\})$, and the output X' should fulfill $\forall c_i \in \{c_1, \dots, c_n\}$.

3.2 Constraints Setups

Our constraints are designed with the following objectives: 1) Following these constraints requires fine-grained and comprehensive capabilities in *Long-In-Long-Out Generation*; 2) The evaluation method for these constraints should be objective and reliable, abandoning the unreliable approach of employing a LLM as a judge to directly score.

Consequently, we avoid constraints such as style, format, and fluency, which are typically scored directly using LLMs in previous text generation benchmarks. Inspired by the NER and RE tasks, and based on the objectives outlined above, we propose five constraints of varying difficulty levels.

C1 Topic Coherence Constraint: Every entity in the generated document X' should be exclusively related to the target topic T' or intended subtopic and maintain contextual coherence. **C2 Length Constraint:** X' should maintain consistency with the original text X in terms of both the number of sentences and the number of words within each sentence. **C3 Sentence Relation Constraint:** The relation between each pair of adjacent sentences in X' should maintain equivalence with the relation between corresponding adjacent sentences in X . **C4 Entity Type Constraint:** Each entity in X' should maintain the same type as its corresponding entity in X . **C5 Entity Relation Constraint:** The relation between each pair of entities in X' should maintain the same type as the relation between the corresponding pair of entities in X .

Interpretation of sentence-relation preservation under topic transfer C3 preserves the discourse function of adjacent-sentence relations rather than their domain-specific lexical content. For exam-

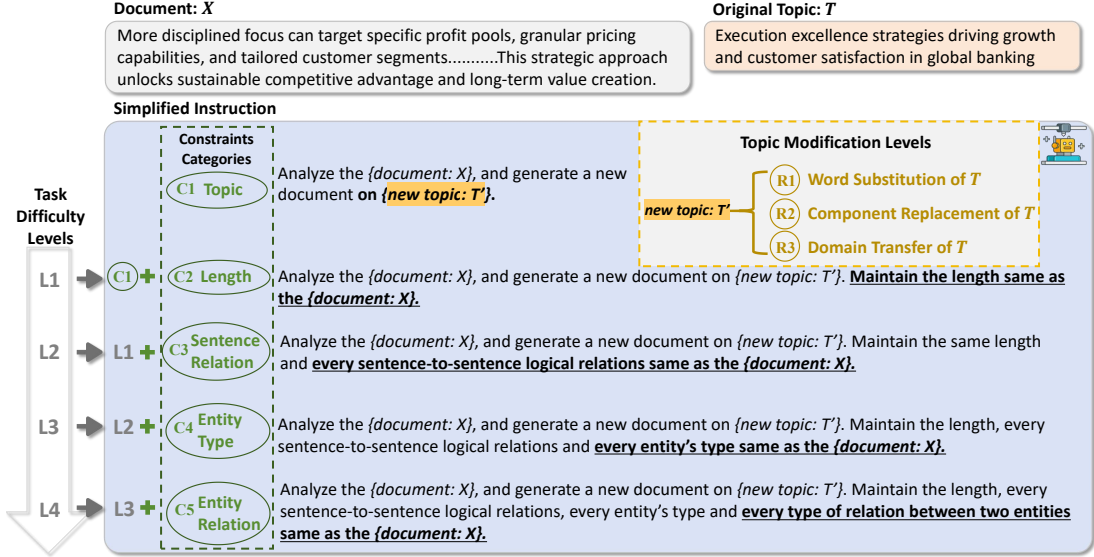


Figure 1: The task framework of *DocFollow*. Task difficulty is defined by Topic Modification Levels (R1–R3) and Difficulty Levels (L1–L4). Difficulty Levels are built cumulatively: L1 (C1 Topic + C2 Length) serves as the baseline, upon which L2 adds C3 (Sentence Relation), L3 adds C4 (Entity Type), and L4 adds C5 (Entity Relation).

ple, if one source sentence states a condition and the next describes its consequence, the rewritten pair should preserve the same conditional relation even when R3 maps the topic to another subdomain. Likewise, causal, contrastive, temporal, and elaborative relations have domain-independent discourse interpretations, although their entities and propositions change with the target topic. R3 transformations remain within one of our five broad domains while changing subdomains, which avoids arbitrary transfer between unrelated communicative settings. During Phase 4, domain experts additionally removed instances for which the transferred topic made the requested relation pattern unnatural or infeasible. We therefore view C3 under R3 as a controlled test of discourse-structure preservation for rewriting tasks, not as a claim that every real-world editing task should preserve every adjacent-sentence relation.

The detailed instruction of each constraint is provided in Appendix B.

3.3 Benchmark Construction

As shown in Figure 2, *DocFollow* is constructed through a structured pipeline that integrates manual document collection, model-driven topic generation and modification with experts reviewing, dataset pre-testing and validation, ensuring the diversity and appropriate difficulties. The whole construction process is detailed below.

Phase 1: Document Collection We established a taxonomic framework comprising five domains, and further divided them into specialized subdomains, yielding a 20-category classification system. A team of 15 experts with professional domain-specific knowledge systematically curated a high-quality, domain-representative corpus for each category. Our document collection drew upon a diverse range of authoritative, open-access repositories to ensure breadth and authenticity. The sources we utilized include multinational bodies like the UN Parallel Corpus, pre-print servers such as arXiv, public-domain libraries like Project Gutenberg, regulatory filings from the SEC EDGAR database, and specialized academic resources including PubMed Central (PMC) and the What Works Clearinghouse (WWC). This multi-source strategy was crucial for capturing the distinct terminologies and stylistic features of each domain (See Appendix C.1 and Appendix C.2 for experts qualifications, IAA results, and full data collection guidelines).

Phase 2: Original Topic Generation We prompt 3 advanced LLMs (ChatGPT-4o (Hurst et al., 2024), Claude-3.5-Sonnet, and DeepSeek-V3 (DeepSeek-AI, 2024)) to generate original topics from collected documents based on designed rules. The model was prompted to generate a single, representative topic per text, constrained to 5–15 words, ensuring precise encapsulation of the core subject. Human experts review the topics generated by the model. If multiple topics meet the requirements for

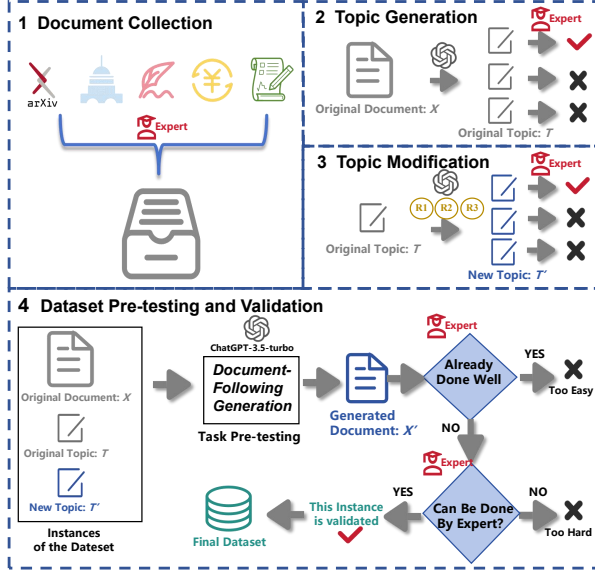


Figure 2: The *DocFollow* data construction pipeline. The process involves (1) Document Collection, (2) Topic Generation, and (3) Topic Modification. The first three phases create initial instances. The final phase (4) Dataset Pre-testing and Validation, validates instances by filtering out those deemed “Too Easy” for an outdated model or “Too Hard” for experts, ensuring appropriate difficulty.

the same document, one is randomly selected to ensure that the data does not originate solely from a single model’s output.

The full protocol, including the specific prompt template and validation checklist, is detailed in Appendix C.3.

Phase 3: Topic Modification To modify the original topic at different levels, we developed a protocol using LLMs to transform original topics into new topics for *Document-Following* tasks. Three transformation levels, progressing in complexity. **R1:** Replaced one or two key content words with antonyms or near-synonyms, subtly altering meaning. This assessed sensitivity to semantic nuances. **R2:** Substituted either the phenomenon or context component with a parallel concept within the same subdomain, preserving syntax. This tested adaptation to new topical constraints. **R3:** Mapped the topic’s argumentative frame to a different subdomain, retaining logical structure, while still in the same domain of five material domains. This required significant topical reorientation. Human experts also review the modified topics. If multiple topics meet the requirements for the same document, one is also randomly selected to ensure that the data does not originate solely from a single

Domain	Subdomain	Tokens			Size
		Max	Min	Avg	
Academic & Engineer	Engineer Report	4388	958	2331	85
	Experiment Design	3641	1448	2260	183
	Research Background	2182	1305	1677	182
	Research Conclusion	2682	829	1735	122
	Technical Design	5445	1223	2853	85
	All Sub Domains	5445	829	2087	657
Education	Case Study	2695	1279	2047	74
	Course Syllabi	1699	776	1329	84
	Edu Management	4669	1715	3236	86
	Reflective Journals	4326	1621	2732	97
	Research Reports	2862	939	1848	134
	All Sub Domains	4669	776	2216	475
Finance & Business	Feasibility Studies	3808	1400	2278	97
	Industry Analysis	3700	1071	2001	122
	Market Research	2276	1094	1826	134
	All Sub Domains	3808	1071	2011	353
Literature & Art	Narrative Scripts	4917	2107	3320	110
	Poetry	8323	750	2971	146
	Short Stories	5284	1372	2889	219
	All Sub Domains	8323	750	3010	475
Politics & Law	AdminComms	5414	1014	2112	170
	Government Reports	4815	1466	2829	110
	Legal Analysis	4572	2075	3074	86
	Procedural Records	5520	1039	2543	110
	All Sub Domains	5520	1039	2543	476
Dataset		8323	750	2610	2436

Table 2: Descriptive statistics of the *DocFollow* dataset. The table details the token count metrics (maximum, minimum, and average) and instance counts (Size) across 5 main domains and their 20 sub-domains.

model’s output.

After getting the X , T and T' that have been verified manually, following the process shown in Figure 1, we constructed the graded difficulty levels of L1 to L4 by gradually increasing the number and complexity of constraints.

Prompts enforced clear output constraints to ensure consistency. This tiered approach created a controlled multi-level difficulty gradient, enabling precise assessment of semantic adaptability across transformation complexities.

The specific prompts for each R1-R3 transformation level and the validation guideline are detailed in Appendix C.4.

Phase 4: Dataset Pre-testing and Validation

We used ChatGPT-3.5-Turbo as a lower-bound difficulty-calibration model rather than as a quality-control model. The purpose of this pre-test was to identify instances that were already trivial for a comparatively weaker model: an instance on which all requested constraints were consistently

satisfied provides limited discrimination among stronger systems. Using the strongest available model for this filtering step would instead risk removing informative instances that remain useful for differentiating current models. Quality and feasibility were determined separately by domain experts. The experts reviewed the pre-test outputs and removed both trivially easy instances and instances whose combined topic transformation and constraints were judged infeasible even for a qualified human writer. This two-sided filtering retains tasks that are challenging for the calibration model while remaining human-feasible. The complete criteria and validation protocol are provided in Appendix C.5.

After filtering, *DocFollow* contains 2,436 instances across five major domains and 20 subdomains, with approximately 120 instances per subdomain (Table 2).

3.4 Evaluation Protocol

Since most constraints are open-ended without reference answers, we propose a novel evaluation method to achieve objective, reliable, and fine-grained assessment. The core innovation lies in decomposing overall evaluation uncertainty into atomic-level classification uncertainties. Specifically, rather than relying on LLMs to directly score constraint adherence, our protocol employs LLMs solely as information extractors for well-established classification tasks—entity typing (NER) and relation classification (RE) (Xu et al., 2024)—while all scoring is computed through deterministic rule-based programs. This decomposition ensures that evaluation reliability depends only on the accuracy of mature, verifiable subtasks, which we empirically validate in Section 5.1.

C1 Topic Coherence Topic Coherence score S_{C1} calculates the average percentage of entities that meet topic coherence standards in the generated document, defined as follows:

$$S_{C1} = 1 - \left(\frac{1}{N'_S} \sum_{i=1}^{N'_S} \frac{N_{inco}^i}{N_W^{i'}} \right) \times 100\%, \quad (1)$$

where N'_S is the number of sentences in X' , N_{inco}^i is the number of the incoherent entities in i -th sentence in X' and $N_W^{i'}$ is the word count of i -th sentence in X' . Incoherent entities are identified by LLMs through one-by-one assessment against

clear and detailed criteria. Each assessment focuses on analyzing only one single entity, thereby reducing demands on LLM capabilities and improving accuracy rates.

C2 Length Evaluation Length score S_{C2} is evaluated directly through calculating the number of sentences and words in X and X' , defined as follows:

$$S_{C2} = \left[1 - \left(a \frac{|N'_S - N_S|}{N_S} + (1 - a) \frac{|N'_W - N_W|}{N_W} \right) \right] \times 100\%, \quad (2)$$

where N_S is the number of sentences in X , N_W and N'_W are the number of words of X and X' . Both the number of sentences and the word count are crucial factors for text length. We use the same instructions for keeping X' have the same number of words and sentences as X , so we set $a = 0.5$.

C3-C5 Sentence Relation, Entity, and Entity Relation

These three constraints follow similar evaluation processes: 1) Map corresponding pairs of sentences relations/entities/entity relations between X and X' ; 2) Identify the types of one pair of mapped sentence relation/entity/entity relation in both X and X' using identical standards; 3) Determining whether the two sentence relations/entities/entity relations in a pair belong to the same type. If so, this pair is labeled as “matched”. The above process is the complete analysis procedure for a single pair of mapping sentence relations/entities/entity relations. Our algorithm traverses every mapped pair, determining the number of “matched” pair. This count is then divided by the total number of sentence relations/entities/entity relations in X' to calculate the final score $S_{C3/C4/C5}$ for C3, C4 and C5, defined as follows:

$$S_{C3/C4/C5} = \frac{N'_{g3/g4/g5}}{N'_{3/4/5}} \times 100\%, \quad (3)$$

where $N'_{g3/g4/g5}$ is the number of “matched” sentence relations/entities/entity relations in X' , and $N'_{3/4/5}$ is the number of total sentence relations/entities/entity relations in X' .

The comprehensive evaluation protocol, including the pseudo-code for our rule-based scoring, the specific prompts used for information extraction, and the predefined classification taxonomies, is detailed in Appendix D.

4 Experiments

4.1 Experiments Settings

We evaluate 13 LLMs including ChatGPT-5.2, ChatGPT-5.1, ChatGPT-4o, Claude-4.5-Sonnet, Claude-4.5-Haiku, LLaMA-3.1-8B (Team, 2024), LLaMA-4-Scout, Qwen3-8B (thinking), Qwen3-8B (no thinking), Qwen3-235B (thinking), Qwen3-235B (no thinking) (Yang et al., 2025), DeepSeek-R1 (DeepSeek-AI, 2025), and DeepSeek-V3.2. We access ChatGPT-5.2, ChatGPT-5.1, ChatGPT-4o via OpenAI API. We access Claude-4.5-Haiku and Claude-4.5-Sonnet via Anthropic API. We access other open-source LLMs from their official repositories. During the evaluation process, we set the temperature to 0 to ensure deterministic outputs for information extractor. Considering that the task is an open-ended writing task, we set the temperature to 0.9 for the tested model. We also conduct an ablation study of the temperature in Appendix G. For each instance, we make the tested model run 3 times and record the average scores.

A complete list of all 13 models, including their developers, specific version identifiers, and access methods, is provided in Appendix F.

4.2 Main Results

The results across three Topic Modification Levels (R1-R3) and four Task Difficulty Levels (L1-L4) are summarized in Table 3 and Table 4, respectively. We also include the discussion on results across different subdomains in Appendix G and case study of common failure patterns in Appendix E.

Granularity-Dependent Performance Degradation A consistent pattern emerges as constraints shift from coarse to fine-grained. As shown in Table 4, models maintain high performance on document-level constraints (S_{C1} : 81.9–92.5%, S_{C2} : 56.6–79.9% at L1–L4), but exhibit significant deterioration on sentence-level relations (S_{C3} : 33.8–62.7%) and entity-level constraints (S_{C4} : 17.5–65.2%, S_{C5} : 16.2–61.3%). This hierarchical degradation persists even in the largest models, including DeepSeek-R1 and Qwen3-235B, indicating an architectural bottleneck rather than a scaling limitation.

Model-Specific Capability Profiles As reported in Table 3, Claude-4.5-Sonnet achieves the highest overall performance (73.7% average), followed by DeepSeek-V3.2 (72.4%). Qwen3-8B (thinking) attains the peak topic coherence (S_{C1} =98.1%

at R2, Table 3) but underperforms on entity constraints (S_{C4} =40.7% at R1, S_{C5} =27.4% at R1, Table 3), suggesting optimization biased toward semantic over structural fidelity. Claude-4.5-Haiku exhibits the most severe entity-level degradation (S_{C4} =29.6% at R2, S_{C5} =22.2% at R2, Table 3), ranking last overall (51.8%). As shown in Table 4, ChatGPT-5.1 shows notably weak length control (S_{C2} : 58.9–61.6% across L1–L4).

Reasoning Model Trade-offs Comparing Qwen3-235B variants in Table 3 reveals a critical trade-off. Without reasoning, the model achieves strong coarse-grained performance on simpler scenarios (S_{C3} =67.3% at R1) but catastrophically fails on fine-grained constraints in complex scenarios (S_{C4} =39.9%, S_{C5} =17.9% at R3). The reasoning-enabled variant sacrifices some coarse-grained performance (S_{C3} =51.6% at R1) but substantially improves complex constraint adherence (S_{C4} =63.2%, S_{C5} =54.7% at R3). This demonstrates that reasoning mechanisms benefit fine-grained control at the cost of efficiency on simpler tasks.

5 Analysis

5.1 Impact of Information Extractor

Our evaluation protocol’s reliability hinges on whether LLMs can accurately perform atomic-level classification tasks. To validate this, we sample 100 instances spanning all constraints and topic modification levels, recruiting 5 experts to serve as information extractors alongside three LLMs. As shown in Table 5, the high ICC scores (Koo and Li, 2016) (average ≈ 0.86) demonstrate strong alignment between LLM and human extractors across all subtasks. This empirically confirms our core methodological claim: by decomposing evaluation into atomic classification tasks, overall reliability reduces to the accuracy of these well-established subtasks—which LLMs perform with human-level consistency. We select DeepSeek-V3 for extraction given its balanced performance and computational efficiency.

5.2 Impact of Input Length

In this section, we analyze the impact of input tokens on constraint adherence across all metrics (S_{C1} - S_{C5}) for three representative models: Claude-4.5-Sonnet, DeepSeek-R1, and ChatGPT-5.1. As illustrated in Figure 3, our analysis reveals a degradation in constraint adherence as input length in-

Model	R1					R2					R3					Avg.
	S_{C1}	S_{C2}	S_{C3}	S_{C4}	S_{C5}	S_{C1}	S_{C2}	S_{C3}	S_{C4}	S_{C5}	S_{C1}	S_{C2}	S_{C3}	S_{C4}	S_{C5}	
Claude-4.5-Sonnet	95.2	79.9	63.8	66.8	58.6	97.1	82.6	65.6	66.6	59.6	91.7	83.2	65.4	67.6	62.5	73.7
Deepseek-V3.2	96.4	79.8	62.2	65.7	57.2	95.3	82.3	66.3	66.8	59.2	91.3	80.3	65.1	65.4	56.6	72.4
DeepSeek-R1 *	90.9	79.3	61.6	68.2	58.2	92.6	82.7	54.4	65.7	60.3	82.7	82.6	61.2	62.8	51.7	70.4
Qwen3-235B (thinking) *	91.1	80.4	51.6	62.7	58.2	91.7	82.6	55.2	63.5	66.3	82.3	84.1	63.4	63.2	54.7	69.9
Qwen3-235B (no thinking)	96.3	79.7	67.3	61.8	61.1	93.8	85.4	63.6	67.8	60.3	91.7	80.6	52.1	39.9	<u>17.9</u>	67.9
ChatGPT-5.2	93.3	69.1	58.1	52.8	48.7	93.5	74.1	66.5	60.7	50.7	92.7	72.2	65.6	54.7	51.7	66.9
ChatGPT-5.1	<u>89.6</u>	<u>60.4</u>	56.3	51.8	49.6	95.2	<u>61.8</u>	63.3	56.5	52.3	92.1	<u>63.6</u>	64.1	55.2	54.8	64.6
LLama 4 Scout	92.6	72.4	52.1	52.6	45.6	91.9	75.7	49.9	52.3	50.6	84.1	76.4	52.8	59.6	49.3	63.9
Qwen3-8B (thinking) *	95.2	72.4	<u>43.6</u>	40.7	27.4	98.1	79.2	47.7	47.6	41.8	92.4	84.6	62.2	59.3	53.7	63.1
Qwen3-8B (no thinking)	95.6	72.8	47.3	41.4	40.2	95.3	75.7	52.4	46.7	45.9	92.3	77.2	54.3	40.8	48.2	61.8
LLaMA-3.1-8B	90.8	67.7	47.1	43.7	31.1	<u>90.6</u>	69.7	46.2	52.1	50.2	88.2	74.2	53.3	49.1	51.4	60.4
ChatGPT-4o	91.9	68.2	44.7	39.4	34.6	93.1	71.3	48.8	44.7	41.3	90.7	74.8	53.8	49.2	38.8	59.0
Claude-4.5-Haiku	90.1	65.6	44.3	<u>34.2</u>	<u>26.2</u>	92.2	66.4	<u>43.3</u>	<u>29.6</u>	<u>22.2</u>	90.7	69.5	<u>45.4</u>	<u>31.3</u>	25.7	<u>51.8</u>

Table 3: Performance of various LLMs across three levels of Topic Modification (R1–R3), each comprising five sub-scores (S_{C1} – S_{C5}). The last column summarizes each model’s average score across all settings. * denotes the “reasoning” model.

Model	L1		L2			L3				L4					Avg.
	S_{C1}	S_{C2}	S_{C1}	S_{C2}	S_{C3}	S_{C1}	S_{C2}	S_{C3}	S_{C4}	S_{C1}	S_{C2}	S_{C3}	S_{C4}	S_{C5}	
Claude-4.5-Sonnet	87.8	76.4	88.8	77.7	60.3	88.2	75.9	61.8	63.4	86.9	79.4	59.3	65.2	61.3	73.7
Deepseek-V3.2	87.7	74.9	87.7	76.6	59.1	87.9	74.8	60.5	61.8	87.2	76.6	57.6	63.2	59.3	72.4
DeepSeek-R1 *	83.7	76.3	83.1	79.7	58.5	<u>82.9</u>	76.6	55.9	60.9	84.7	79.1	49.5	61.2	53.6	70.4
Qwen3-235B (thinking) *	84.4	76.4	83.9	78.1	49.8	<u>84.6</u>	78.1	54.8	56.5	83.6	79.2	54.1	60.2	55.2	69.9
Qwen3-235B (no thinking)	89.1	77.4	90.2	79.9	58.7	91.1	76.8	53.7	49.7	89.8	76.9	58.6	53.9	42.9	67.9
ChatGPT-5.2	88.6	71.1	89.4	64.8	57.1	89.3	63.5	58.7	49.6	88.7	<u>58.6</u>	59.7	52.4	45.9	66.9
ChatGPT-5.1	92.5	60.3	91.3	58.9	62.7	89.7	58.9	60.6	52.3	91.3	61.6	54.3	50.8	51.7	64.6
LLaMa 4 Scout	84.7	67.8	83.2	70.3	48.9	84.3	69.7	46.5	50.7	84.8	71.7	46.2	49.2	44.7	63.9
Qwen3-8B (thinking) *	87.5	71.1	87.3	66.9	39.5	88.9	74.1	48.3	42.1	87.2	72.7	44.8	42.8	35.5	63.1
Qwen3-8B (no thinking)	88.2	65.8	88.2	68.8	43.2	87.2	70.3	46.4	33.5	87.3	70.1	45.7	39.3	37.4	61.8
LLaMA-3.1-8B	83.3	65.4	85.4	63.2	41.1	83.4	64.3	40.2	42.4	84.6	63.3	44.9	43.6	37.9	60.4
ChatGPT-4o	84.7	59.8	84.5	64.3	42.9	84.5	66.3	44.1	35.7	84.9	66.4	42.8	40.1	33.4	59.0
Claude-4.5-Haiku	<u>82.8</u>	<u>56.6</u>	<u>82.7</u>	<u>58.4</u>	<u>33.8</u>	83.2	<u>58.8</u>	<u>34.4</u>	<u>17.5</u>	<u>81.9</u>	59.2	<u>35.1</u>	<u>22.0</u>	<u>16.2</u>	<u>51.8</u>

Table 4: Performance of various LLMs across four levels of difficulty levels (L1–L4), each comprising an increasing number of sub-scores (S_{C1} – S_{C5}). The last column summarizes each model’s average score across all settings. * denotes the “reasoning” model.

Evaluator	S_{C1}	S_{C3}	S_{C4}	S_{C5}	Avg
ChatGPT-4o	0.91	0.88	0.87	<u>0.81</u>	0.87
DeepSeek-V3	0.90	<u>0.85</u>	0.86	0.83	0.86
Claude-3.5-Sonnet	<u>0.89</u>	0.87	<u>0.84</u>	<u>0.81</u>	<u>0.85</u>

Table 5: ICC scores across four tasks when human experts and three LLMs serve as information extractors on the same 100 instances.

creases, consistent with a hypothesis of attentional decay leading to errors in long-form generation. We can also find 3000 tokens represents a critical threshold where performance on S_{C1} , S_{C2} begins to degrade rapidly. In contrast, for more fine-grained constraints (S_{C1} – S_{C5}), the decline is more steady across all input lengths. This suggests that as context increases, models struggle with fine-grained details even in shorter documents, and their ability to manage document-level controlling starts fade in longer input.

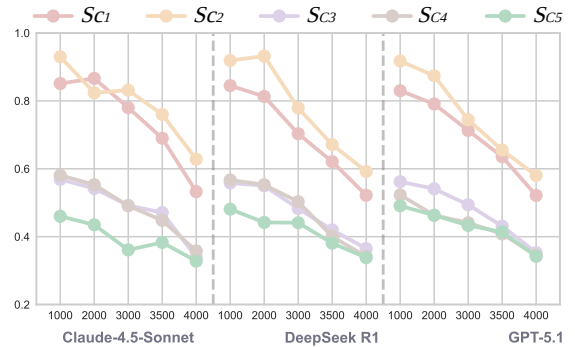


Figure 3: Performance degradation on five constraints (S_{C1} – S_{C5}) as input token length increases.

5.3 Further Analysis in Appendix

We developed *DocFollower*, a graph-guided generation framework that explicitly extracts entity-relation structures to guide step-by-step text generation. Despite achieving 10.9% improvement over baselines, entity-level constraints (S_{C4} , S_{C5}) re-

mained below 70%, confirming that current LLMs exhibit systematic weaknesses in fine-grained control even with structured guidance. Additional analyses—including baseline comparisons (CoT, Few-shot), positional constraint adherence patterns, and human correlation validation—are detailed in Appendix A–H.

6 Conclusion

We introduce *DocFollow*, the first comprehensive benchmark for evaluating LLMs’ ability to follow hierarchical fine-grained constraints in *Long-In-Long-Out Generation* tasks. Through 2,436 instances evaluation, we systematically assess 13 models across increasing levels of constraint granularity. Our results reveal a consistent pattern: while models maintain reasonable performance on coarse-grained constraints, they exhibit substantial degradation on fine-grained requirements, particularly entity-level constraints. This hierarchical performance suggests fundamental limitations in current LLMs’ ability to maintain detailed structural and semantic fidelity throughout extended generation. These findings highlight the critical need for innovations beyond scaling to achieve precise controllability in long-form text generation.

Limitations

Despite our study’s contributions, *DocFollow* exhibits several limitations that should guide its interpretation and motivate future work. First, our benchmark, though comprehensive, centers exclusively on text rewriting tasks. Though rewriting inherently encompasses many cognitive complexities of natural language generation, future work should explore extending this methodology to other generation paradigms such as ab initio creation, summarization, or translation. Expanding language coverage and incorporating naturally occurring documents from a broader spectrum of registers would strengthen the external validity of our conclusions.

Second, although we collected 2,436 instances spanning five major domains and twenty subdomains to ensure coverage of diverse professional contexts, the benchmark remains limited to English-language texts and curated subdomains. Consequently, its findings may not generalize to non-English corpora, informal or conversational registers, or highly specialized domains (e.g., legal statutes with jurisdictional variability).

Third, computational resources required for com-

prehensive evaluation across multiple constraint levels and topic modification depths may present accessibility barriers for researchers with limited computational resources. Our benchmark’s complexity necessitates significant computational investment, particularly when evaluating multiple runs with varying parameters. Future work should explore fine-tuning smaller, specialized models specifically for the information extraction components of our evaluation pipeline, which could substantially reduce computational costs while maintaining evaluation accuracy.

Ethics Statement

We acknowledge that the text generation capabilities evaluated in *DocFollow* could potentially be misused for generating deceptive or misleading content at scale. Sophisticated text transformation skills might enable the creation of counterfeit documents or facilitate the repurposing of legitimate content for harmful purposes. Therefore, maintaining responsible research practices when developing and deploying text rewriting models is of paramount importance. In *DocFollow*, all evaluation instances undergo thorough domain expert validation to ensure they represent reasonable rewriting challenges without encouraging harmful applications. This deliberate approach underscores our commitment to advancing text generation capabilities within an ethical framework that prioritizes beneficial applications while minimizing potential misuse.

Acknowledgments

This work is supported by Advanced Materials-National Science and Technology Major Project (Grant No. 2025ZD0620100), National Key R&D Program of China (No. 2024YFA1012700), Yangcheng Scholars Research Project (No.2024312049), and Guangdong Provincial Key Lab of Integrated Communication, Sensing and Computation for Ubiquitous Internet of Things (No. 2023B1212010007).

During the preparation of this work, the authors used LLMs for assistance with language polishing and literature survey. The core research design, experimental analysis, and intellectual contributions were conducted solely by the authors, who take full responsibility for all content presented in this paper.

References

- Chenxin An, Shansan Gong, Ming Zhong, Xingjian Zhao, Mukai Li, Jun Zhang, Lingpeng Kong, and Xipeng Qiu. 2024. L-eval: Instituting standardized evaluation for long context language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14388–14411.
- Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024. [Longbench: A bilingual, multi-task benchmark for long context understanding](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, ACL 2024, Bangkok, Thailand, August 11-16, 2024, pages 3119–3137. Association for Computational Linguistics.
- Yushi Bai, Jiajie Zhang, Xin Lv, Linzhi Zheng, Siqi Zhu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2025. Longwriter: Unleashing 10,000+ word generation from long context llms. In *International Conference on Learning Representations*, volume 2025, pages 36528–36546.
- Cheng-Han Chiang and Hung-yi Lee. 2023. Can large language models be an alternative to human evaluations? In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15607–15631.
- DeepSeek-AI. 2024. [Deepseek-v3 technical report](#). CoRR, abs/2412.19437.
- DeepSeek-AI. 2025. [Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning](#). CoRR, abs/2501.12948.
- Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*.
- Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, Yuanzhuo Wang, and Jian Guo. 2024. [A survey on llm-as-a-judge](#). CoRR, abs/2411.15594.
- Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.
- Yuxin Jiang, Yufei Wang, Xingshan Zeng, Wanjuan Zhong, Liangyou Li, Fei Mi, Lifeng Shang, Xin Jiang, Qun Liu, and Wei Wang. 2024. Follow-bench: A multi-level fine-grained constraints following benchmark for large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4667–4688.
- Terry K Koo and Mae Y Li. 2016. A guideline of selecting and reporting intraclass correlation coefficients for reliability research. *Journal of chiropractic medicine*, 15(2):155–163.
- Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, 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](#). In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.
- Haoran Que, Feiyu Duan, Liqun He, Yutao Mou, Wangchunshu Zhou, Jiaheng Liu, Wenge Rong, Zekun Moore Wang, Jian Yang, Ge Zhang, et al. 2024. Hellobench: Evaluating long text generation capabilities of large language models. *arXiv preprint arXiv:2409.16191*.
- Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. *Advances in neural information processing systems*, 36:53728–53741.
- Rickard Stureborg, Dimitris Alikaniotis, and Yoshi Suhara. 2024. Large language models are inconsistent and biased evaluators. *arXiv preprint arXiv:2405.01724*.
- Haochen Tan, Zhijiang Guo, Zhan Shi, Lu Xu, Zhili Liu, Yunlong Feng, Xiaoguang Li, Yasheng Wang, Lifeng Shang, Qun Liu, et al. 2024. Proxyqa: An alternative framework for evaluating long-form text generation with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6806–6827.
- Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models*. <https://crfm.stanford.edu/2023/03/13/alpaca.html>, 3(6):7.

- Llama Team. 2024. [The llama 3 herd of models](#). *CoRR*, abs/2407.21783.
- Hengyi Wang, Haizhou Shi, Shiwei Tan, Weiyi Qin, Wenyan Wang, Tunyu Zhang, Akshay Nambi, Tanuja Ganu, and Hao Wang. 2025. Multimodal needle in a haystack: Benchmarking long-context capability of multimodal large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3221–3241.
- Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Lingpeng Kong, Qi Liu, Tianyu Liu, et al. 2024. Large language models are not fair evaluators. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 9440–9450.
- Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.
- Yuhao Wu, Ming Shan Hee, Zhiqiang Hu, and Roy Ka-Wei Lee. 2025. Longgenbench: Benchmarking long-form generation in long context llms. In *International Conference on Learning Representations*, volume 2025, pages 6851–6872.
- Yuning Wu, Jiahao Mei, Ming Yan, Chenliang Li, Shaopeng Lai, Yuran Ren, Zijia Wang, Ji Zhang, Mengyue Wu, Qin Jin, et al. 2026. Writingbench: A comprehensive benchmark for generative writing. *Advances in Neural Information Processing Systems*, 38.
- Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, Yang Wang, and Enhong Chen. 2024. Large language models for generative information extraction: A survey. *Frontiers of Computer Science*, 18(6):186357.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Xinrong Zhang, Yingfa Chen, Shengding Hu, Zihang Xu, Junhao Chen, Moo Khai Hao, Xu Han, Zhen Leng Thai, Shuo Wang, Zhiyuan Liu, and Maosong Sun. 2024. [\$\infty\$ bench: Extending long context evaluation beyond 100k tokens](#). *CoRR*, abs/2402.13718.
- Zikang Zhang, Wangjie You, Tianci Wu, Xinrui Wang, Juntao Li, and Min Zhang. 2025. A survey of generative information extraction. In *Proceedings of the 31st International conference on computational linguistics*, pages 4840–4870.
- Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhaghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

A *DocFollower*: An Entity-Relation Graph Framework for Controllable Generation

To further validate the findings of our benchmark, we propose *DocFollower*, a framework that demonstrates how explicit entity-relation modeling can improve fine-grained constraint adherence in long-form generation tasks. *DocFollower* employs a two-stage approach: 1) Entity-Relation Graph Building and 2) Graph Mapping and Generation, as illustrated in Figure 4.

A.1 Stage 1: Entity-Relation Graph Building

The first stage focuses on extracting and organizing the structural skeleton of the original document X through iterative graph decomposition:

Entity-Relation Extraction *DocFollower* begins by parsing the original document X to extract entities and their relationships, storing them as triples (e_i, r_{ij}, e_j) where e_i, e_j are entities and r_{ij} represents their relationship. This creates a comprehensive entity-relation graph $G = (V, E)$ where vertices V represent entities and edges E represent relationships.

Peripheral Entity Ranking To identify the core narrative structure, *DocFollower* employs a graph-based ranking algorithm that computes entity centrality scores. Specifically, we identify peripheral entities as those connected by the 10% of relations with the lowest edge degrees in G . This approach ensures that we preserve the most structurally significant entities.

Iterative Core Extraction Through multiple iterations, *DocFollower* progressively removes peripheral entities and their associated relations, creating a hierarchy of document representations $\{G_1, G_2, \dots, G_k\}$ where $G_i \subset G_{i-1}$. Each iteration yields a corresponding core outline C_i that captures the document’s essence at different levels of granularity. This hierarchical decomposition enables fine-grained control over the generation process.

A.2 Stage 2: Graph Mapping and Generation

The second stage transforms the extracted structure according to the target constraints while preserving

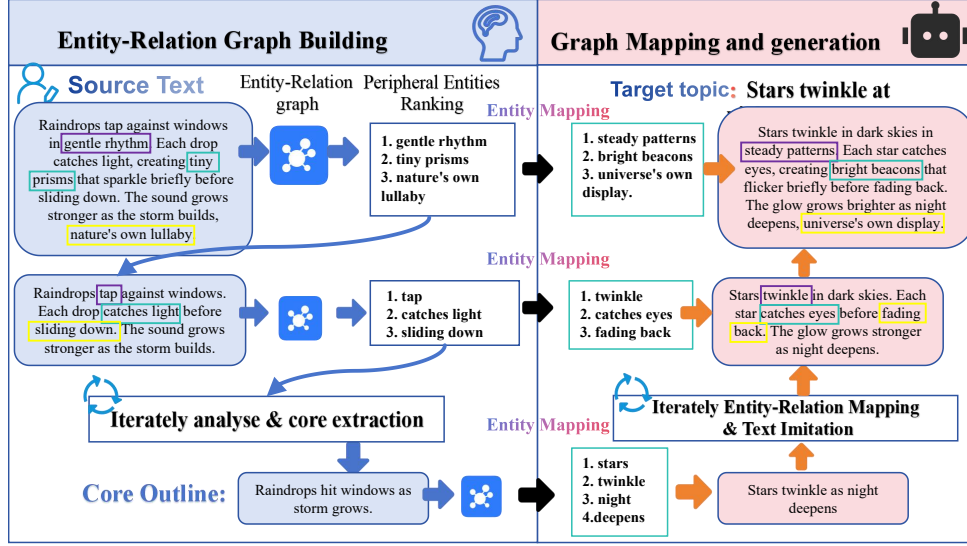


Figure 4: Overview of the *DocFollower* framework for controllable *Document-Following* Generation. The framework operates in two stages: (1) **Entity-Relation Graph Building** (left, blue) extracts entities and relationships from the original document about raindrops, iteratively identifies peripheral entities through graph analysis, and generates hierarchical core outlines of varying granularity; (2) **Graph Mapping and Generation** (right, pink) transforms the extracted structure according to the modified topic constraint (stars), establishing type-preserving entity mappings (e.g., "raindrops" → "stars", "catches light" → "catches eyes") and iteratively generating text that maintains the original’s structural and relational patterns. The colored highlights in the text indicate corresponding entity mappings across the transformation. This two-stage approach enables fine-grained control over entity-level constraints while preserving document structure throughout the generation process.

entity-level and relational fidelity:

Entity Type-Preserving Mapping Given a modified topic T' and constraint set $\mathcal{C}$, *DocFollower* establishes a mapping function $f : V \rightarrow V'$ that maps each entity in the original graph to a semantically appropriate entity in the new document’s context. Crucially, this mapping preserves entity types—for instance, mapping “raindrops” to “stars” maintains the entity category of natural phenomena while adapting to the new context.

Relation Type Preservation Beyond entity mapping, *DocFollower* ensures that relationship types remain consistent. If $(e_i, r_{ij}, e_j) \in G$, then $(f(e_i), r'_{ij}, f(e_j)) \in G'$ where r'_{ij} preserves the semantic relationship type of r_{ij} . For example, the “catches light” relationship between “raindrops” and “light” transforms to “catches eyes” between “stars” and “eyes,” maintaining the semantic interaction pattern.

Iterative Generation with Graph Guidance

The text generation process follows a coarse-to-fine approach, starting with the most abstract core outline C_k and progressively incorporating details from $C_{k-1}, \dots, C_1$. At each iteration i , *DocFollower*:

1. Maps entities from G_i to create G'_i based on the target constraints
2. Generates text segments that instantiate the mapped entity-relation triples
3. Validates that generated text maintains the required constraint adherence
4. Integrates the new content with previously generated text

This iterative process ensures that the final document Y satisfies all five constraint types: 1) length requirements through controlled expansion, 2) topic coherence through consistent entity mapping, 3) logical flow through preserved relationship structures, 4) entity type consistency through type-preserving mappings, and 5) relation type fidelity through semantic relationship preservation.

Experimental Results Table 6 evaluates *DocFollower* against five baseline methods across three state-of-the-art LLMs. Key findings include:

DocFollower Consistently Outperforms Baselines: *DocFollower* achieves the highest average performance across all models (ChatGPT-4.1: 77.1%, DeepSeek-V3: 77.5%, Claude-3.5-Sonnet:

Model	R1					R2					R3					Avg.
	S_{C1}	S_{C2}	S_{C3}	S_{C4}	S_{C5}	S_{C1}	S_{C2}	S_{C3}	S_{C4}	S_{C5}	S_{C1}	S_{C2}	S_{C3}	S_{C4}	S_{C5}	
ChatGPT-4.1																
<i>DocFollower</i>	94.7	89.2	73.3	68.3	65.7	93.2	88.7	62.9	68.2	68.3	93.6	82.0	73.6	67.1	68.2	77.1
Direct	92.9	77.2	57.9	62.4	51.2	93.9	78.4	60.2	62.4	54.9	84.2	75.0	60.9	61.2	52.6	68.4
CoT	88.2	71.3	52.6	60.8	48.0	88.7	76.1	52.1	60.6	44.5	81.9	77.8	54.7	58.5	53.4	64.6
Self Critic	92.9	78.3	58.1	60.5	49.3	97.8	84.7	57.8	65.9	58.0	83.1	78.6	60.9	62.9	57.5	69.8
One shot	94.8	78.1	61.0	68.1	51.6	94.3	79.4	64.4	63.5	56.9	82.1	75.2	64.4	59.8	52.8	69.8
Few shot	94.5	75.8	59.2	63.5	55.1	92.6	86.9	59.6	65.9	62.7	86.5	75.6	66.1	62.6	60.7	71.8
DeepSeek-V3																
<i>DocFollower</i>	94.7	84.3	74.6	68.8	67.8	94.0	80.7	71.0	63.6	64.6	96.5	88.4	75.1	69.0	68.7	77.5
Direct	91.0	71.5	51.3	51.4	44.4	91.5	73.1	48.5	52.0	49.7	82.4	76.7	52.9	58.5	49.5	63.0
CoT	89.9	65.4	49.5	48.2	37.8	90.8	69.9	50.7	42.9	42.1	82.3	67.7	43.8	52.9	48.3	58.8
Self Critic	92.1	72.9	54.0	54.2	52.5	92.3	82.4	47.5	63.0	55.5	87.2	75.9	53.6	51.2	55.3	66.0
One shot	89.4	76.1	49.9	51.2	42.9	94.7	73.8	49.7	52.5	53.0	80.6	81.3	52.6	60.2	51.1	63.9
Few shot	97.4	71.4	50.4	57.4	54.0	99.3	73.0	49.7	53.6	49.5	83.1	74.7	59.1	65.0	59.4	66.5
Claude-3.5-Sonnet																
<i>DocFollower</i>	93.5	77.2	69.3	52.4	58.7	94.6	82.4	64.2	61.3	67.3	95.4	87.9	79.4	67.6	64.6	74.4
Direct	92.1	68.0	42.6	32.1	21.2	91.3	70.0	43.7	32.8	25.1	91.5	73.5	48.7	35.0	34.6	53.5
CoT	89.1	69.2	39.0	26.5	17.2	91.3	66.6	44.8	31.0	16.9	91.7	73.9	49.0	35.8	32.1	51.6
Self Critic	90.8	63.9	40.6	25.7	22.0	91.7	65.7	39.6	25.0	25.7	90.8	67.3	47.3	35.7	36.2	51.2
One shot	95.1	66.0	42.3	29.8	19.6	91.9	67.5	42.4	38.0	26.6	94.4	78.1	51.3	34.1	39.3	54.4
Few shot	97.3	77.7	45.9	39.0	25.1	92.2	78.4	49.4	35.7	28.6	95.4	78.1	46.3	38.6	42.4	58.0

Table 6: Performance comparison of *DocFollower* against five baseline methods (Direct, CoT, Self Critic, One-shot, and Few-shot) across three state-of-the-art LLMs on the *DocFollow* benchmark. *DocFollower* consistently outperforms all baselines, though performance universally degrades from coarse-grained (S_{C1} - S_{C2}) to fine-grained constraints (S_{C4} - S_{C5}), highlighting fundamental challenges in entity-level control for long-form generation.

74.4%), representing a 10.9% improvement over the best baseline methods.

1) Systematic Performance Degradation: All methods exhibit consistent performance drops from coarse-grained constraints (S_{C1} : 89.1-97.4%) to fine-grained entity-level constraints (S_{C5} : 17.2-68.7%). *DocFollower* mitigates this degradation but cannot eliminate it—no model exceeds 70% on S_{C5} , even with explicit graph guidance. 2) Baseline Limitations: Chain-of-Thought surprisingly underperforms (avg. 51.6-64.6%), while few-shot learning fails to maintain entity-level precision despite competitive coarse-grained performance. These results confirm that neither reasoning steps nor examples alone can ensure fine-grained constraint adherence. 3) Fundamental Architectural Constraints: Despite *DocFollower*’s improvements, entity relation preservation S_{C4} and entity-level constraints S_{C5} remain challenging across all models, suggesting that external scaffolding cannot fully compensate for architectural limitations in maintaining structural fidelity throughout extended generation.

By forcing models to explicitly track and transform entity-relation structures, *DocFollower* demonstrates that targeted architectural interven-

tions can significantly improve performance on fine-grained constraints—validating both the importance of our benchmark and pointing toward promising directions for future research in controllable long-form generation. The specific prompts can be found in Figures 40 to 49, and the specific codes are in the supplementary material.

B Constraints details

This section elaborates on the five constraints introduced in Section 3.2. For each constraint, we provide a formal definition and reference the corresponding figure that illustrates its implementation, including the specific prompt template and examples.

B.1 General Prompt Framework

Figure 18 presents the foundational prompt framework that serves as the basis for all experimental tasks. The template defines the core structure of the "Document-Following" task, accepting an original document and a target topic as primary inputs, and incorporating modular placeholders for the five constraints (C1-C5).

B.2 C1: Topic Coherence Constraint

This constraint ensures the generated text X' is thematically coherent with the target topic T' . The specific prompt instructions for enforcing this constraint, including the need to adjust topic-specific vocabulary, are detailed in Figure 19.

B.3 C2: Length Constraint

This constraint governs the length of the generated text X' , requiring it to maintain structural parity with the source text X . The specific instructions provided to the model, detailing requirements for both sentence and word counts, are presented in Figure 20.

B.4 C3: Sentence Relation Constraint

This constraint governs inter-sentence relationships, requiring that the logical and semantic relation between any adjacent sentence pair in the generated text X' remains equivalent to the corresponding pair in the source text X . The specific instructions and examples used to enforce this rule are presented in Figure 21.

B.5 C4: Entity Type Constraint

This constraint focuses on named entities, requiring that each entity in the generated text X' preserves the same type (e.g., Person, Organization) as its corresponding entity in the source text X . The detailed prompt instructions for enforcing this rule are presented in Figure 22.

B.6 C5: Entity Relation Constraint

This constraint governs the relationships between named entities, mandating that the semantic relation between any pair of corresponding entities in the generated text X' remains identical to that in the source text X . The full prompt for enforcing this complex constraint is detailed in Figure 23.

C Dataset Construction Pipeline

This appendix section elaborates on the methodologies for dataset construction and validation. We detail the protocol for expert recruitment, the multi-phase data construction pipeline, and the human evaluation study used to validate our automatic metrics.

C.1 Recruitment and Qualification of Domain Experts

To ensure the high quality of our data curation and validation processes, our study involved a team of

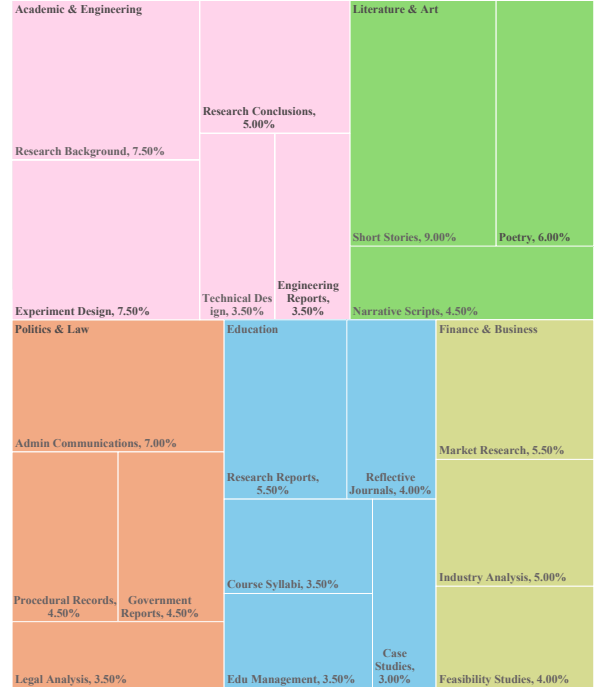


Figure 5: Statistical information of different domains in *DocFollow*.

15 domain experts. All experts were senior PhD students, possessing a minimum of three years of intensive research or professional experience in their designated fields. These experts were organized into five distinct three-person teams, with each team assigned to one of the five primary domains. Prior to the main task, all candidates underwent a rigorous qualification test designed to assess their ability to evaluate writing quality and identify subtle structural differences between texts. Upon selection, all participants signed confidentiality agreements.

To quantitatively measure and ensure expert consistency across the entire data construction pipeline, we conducted a comprehensive Inter-Annotator Agreement (IAA) study. This study was implemented at three critical manual validation junctures: 1) following the Original Topic Generation, 2) after the Topic Modification, and 3) during the Validation and Dataset Pre-testing. For each of these three validation stages, a distinct set of 50 instances per domain was independently assessed by the three experts in the corresponding team. We calculated the agreement on their accept/reject decisions using Fleiss' Kappa. The resulting scores for the three stages were $\kappa_1 = 0.88$ for stage 1, $\kappa_2 = 0.85$ for stage 2, and $\kappa_3 = 0.82$ for stage 3, respectively. These consistently high scores all indicate "substantial agreement," confirming the reliability of

our expert-driven judgments throughout the entire pipeline.

Experts were compensated at a competitive rate of \$25 per hour for their contributions. The average engagement time was 25-30 hours per person, resulting in an approximate total compensation of \$625-\$750 per evaluator. To further incentivize thoroughness, additional bonus of \$100 was provided to evaluators whose total contribution exceeded a pre-defined workload threshold.

C.2 Document Collection

In this initial phase, our five domain experts curated a high-quality corpus for each of the 20 subdomains. The selection process was not arbitrary but followed a unified, rigorous standard to ensure all collected texts demonstrated a high degree of domain authenticity, representativeness, and thematic centrality. The complete set of principles, technical requirements, and sourcing policies that each expert adhered to is detailed in our official Data Collection Guideline (see Figure 8 for the full guideline).

Figure 5 visually demonstrates the distribution of our dataset across various subdomains. The data is almost evenly distributed, with each of the 20 subdomains containing approximately 120 instances. This balanced distribution ensures our benchmark provides thorough coverage across diverse domains, enabling detailed analysis of model performance in various contexts. This multi-domain benchmark supports cross-domain text generation evaluation, and provides reference data for analyzing model performance across professional contexts in multi-domains with varying text lengths and diverse context.

The following are the established, peer-reviewed, and institutionally maintained repositories from which our corpus was constructed. All data sources have been verified for security compliance and contain only publicly available materials:

- [1] United Nations Office at Geneva. (2016). *UN Parallel Corpus*. Geneva: United Nations.
- [2] Ginsparg, P. (1991). *arXiv e-Print Archive*. Ithaca, NY: Cornell University Library.
- [3] Hart, M. S. (1971). *Project Gutenberg*. Urbana, IL: Project Gutenberg Literary Archive Foundation.
- [4] U.S. Securities and Exchange Commission.

(1996). *EDGAR Database*. Washington, DC: U.S. Securities and Exchange Commission.

- [5] U.S. National Library of Medicine. (2000). *PubMed Central*. Bethesda, MD: National Center for Biotechnology Information.
- [6] Institute of Education Sciences. (2002). *What Works Clearinghouse*. Washington, DC: U.S. Department of Education.

C.3 Original Topic Generation

Following the corpus collection, we employed three large language models (LLMs) to generate a representative topic for each document. This process was standardized through a universal prompt template, which instructed the models to distill the core essence of a text into a concise phrase of 5-15 words. The specific prompt template used to guide the LLMs is detailed in Figure 24. Subsequently, each generated topic underwent a rigorous manual review by our domain experts, who followed a comprehensive checklist to validate its quality and relevance (see Figure 9 for the full guideline).

C.4 Topic Modification

To create a controlled difficulty gradient for our benchmark, original topics were systematically transformed into new target topics using our R1-R3 protocol. Each transformation level was executed using a distinct set of prompts designed to manipulate the topic’s semantic and structural properties in a controlled manner. The specific prompts and concrete examples for each of the R1, R2, and R3 transformations are detailed in Figure 25. Furthermore, to ensure the quality and validity of each transformation, the generated topics were manually reviewed against a rigorous checklist, which is provided in our validation guideline (see Figure 10 for the full guideline).

C.5 Dataset Pre-testing and Validation

The final phase ensured the quality and appropriate difficulty of all constructed instances. This involved a two-stage process: an initial large-scale pre-test with ChatGPT-3.5-Turbo, followed by a thorough manual review by our domain specialists. The expert review process was structured by a clear validation protocol, which included specific criteria for identifying and filtering out overly simple or prohibitively difficult instances. The detailed validation criteria are outlined in the Data Collection Guideline (see Figure 11 for the full guideline).

D Evaluation details

D.1 Overview of the Evaluation

Our evaluation protocol is designed to be objective, reliable, and fine-grained, moving away from unreliable LLM-as-a-judge scoring. The core principle is to employ Large Language Models (LLMs) strictly as information extractors for simple, targeted classification tasks. The structured information extracted by the LLM is then fed into deterministic, rule-based programs that calculate the final scores for each constraint (S_{C1} - S_{C5}). This two-step process ensures that the final judgment is based on objective rules rather than the opaque reasoning of a scoring model.

To illustrate this process, we present the complete algorithm for calculating the Entity Relation Score (S_{C5}) as a representative example in Algorithm 1. While the specific extraction and comparison rules differ for other constraints (S_{C1} , S_{C3} , S_{C4}), they all follow this fundamental principle of using an LLM for structured data extraction followed by rule-based scoring. The Length Score (S_{C2}) is calculated directly from the text without LLM involvement.

Algorithm 1 Entity Relations Evaluation

Require: $T_O, T_R, \mathcal{M}, \mathcal{D}$

Ensure: Score $S \in [0, 1]$

```

1:  $d \leftarrow \text{CLASSIFYDOMAIN}(T_O, T_R, \mathcal{M})$ 
2:  $O, R \leftarrow \text{SPLITSENT}(T_O, T_R)$ 
3:  $E_o, E_r \leftarrow \text{EXTRACTENT}(O, R, d, \mathcal{M})$ 
4:  $A \leftarrow \text{GETSENTALIGN}(O, R, \mathcal{M})$ 
5:  $S_{tot} \leftarrow 0, n \leftarrow 0$ 
6: for  $(i_r, i_o) \in A$  do
7:    $s_o \leftarrow O[i_o], s_r \leftarrow R[i_r]$ 
8:    $e_o \leftarrow E_o[i_o], e_r \leftarrow E_r[i_r]$ 
9:    $T \leftarrow \text{GETRELTAX}(d, \mathcal{D})$ 
10:   $L_o \leftarrow \text{EXTRACTREL}(s_o, e_o, T, \mathcal{M})$ 
11:   $L_r \leftarrow \text{EXTRACTREL}(s_r, e_r, T, \mathcal{M})$ 
12:   $P_o \leftarrow \{(s, r, o) \mid (s, r, o) \in L_o\}$ 
13:   $P_r \leftarrow \{(s, r, o) \mid (s, r, o) \in L_r\}$ 
14:   $M \leftarrow \text{TRIPLEMAP}(s_o, s_r, P_o, P_r, \mathcal{M})$ 
15:   $m \leftarrow |\{(r, o) \in M \mid P_o[o].r = P_r[r].r\}|$ 
16:   $S_{cur} \leftarrow m/|P_r|$ 
17:   $S_{tot} \leftarrow S_{tot} + S_{cur}, n \leftarrow n + 1$ 
18: end for
19: return  $S_{tot}/n$ 

```

D.2 Instructions for Information Extractors

To perform the information extraction tasks outlined in our evaluation protocol, we provide the Large Language Model (LLM) with a series of clear, templated prompts. Each prompt is designed as a simple classification task focused on a single linguistic element to maximize accuracy and reliability. The specific prompts used for evaluating constraints S_{C1} , S_{C3} , S_{C4} , and S_{C5} are detailed below, with full prompt text provided in the corresponding figures.

Evaluation Protocol for Topic Coherence (S_{C1})

For the S_{C1} score, the LLM evaluates a given sentence against the target topic and its surrounding context. The prompt requires a multi-faceted assessment across three distinct criteria: relevance, appropriateness, and coherence. For each criterion, the model provides a binary decision (Yes/No) and supporting details, with the final output structured in a JSON format. The complete prompt template is presented in Figure 29.

Evaluation Protocol for Sentence Relation (S_{C3})

The evaluation for the S_{C3} score involves a two-step protocol using distinct LLM prompts. The first step uses the prompt in Figure 31 to align sentences between the original and generated texts. Subsequently, the second step uses the prompt in Figure 30 to classify the logical relationship of each aligned sentence pair, selecting a type from the taxonomy shown in Figure 26.

Evaluation Protocol for Entity Type (S_{C4})

Similar to the S_{C3} evaluation, the protocol for the S_{C4} score is a two-step process using distinct LLM prompts. The first step employs the prompt in Figure 33 to map corresponding entities between the original and generated texts. Subsequently, the second step uses the prompt detailed in Figure 32 to classify the ontological type of each entity in both texts, selecting a type from the taxonomy shown in Figure 27.

Evaluation Protocol for Entity Relation (S_{C5})

As the most granular evaluation, the protocol for the S_{C5} score mirrors the two-step process of S_{C4} . First, the prompt in Figure 34 is used to perform a Relation Extraction task, identifying all entity relation triples (e.g., subject-relation-object) within both the original and generated texts. This extraction is based on the domain-specific taxonomies presented in Figure 28. Subsequently, a second

prompt, detailed in Figure 35, is employed to map the corresponding relation triples between the two texts for the final rule-based comparison.

E Case Study

To systematically evaluate LLM performance in document following tasks, we conducted a comprehensive analysis of actual model-generated outputs to identify systematic failure patterns. Our methodology involved examining model responses across various document following scenarios and analyzing the specific errors that emerged from these attempts.

Through detailed examination of model outputs, we observed recurring issues that manifested across different texts and transformation tasks. From this analysis, we identified and categorized four primary error types that consistently appeared: 1) **Thematic Inconsistency Errors**, where models failed to maintain coherent thematic vision throughout the text (see Figure 12); 2) **Structural Preservation Failures**, where essential organizational elements were inappropriately modified or eliminated (see Figure 13); 3) **Logical Reasoning Breakdowns**, where models violated basic temporal, causal, or conditional logic relationships (see Figure 14), and 4) **Information Completeness Errors**, where critical details were omitted or irrelevant information was inappropriately added (see Figure 15).

For each error category that emerged from our analysis, we selected representative examples from the model’s actual responses that best illustrate these systematic problems. These examples demonstrate clear, unambiguous failures requiring minimal background knowledge to identify, showing how the same types of errors manifest consistently across different document following contexts.

Our analysis presents problematic transformations across various complex document following tasks. Figures 12 through 15 provide detailed examples of each error type, demonstrating the systematic nature of these failures across different textual domains.

As illustrated in the accompanying figures, these systematic failure patterns discovered through empirical analysis reveal fundamental limitations in current LLM architectures’ ability to maintain constraint satisfaction, global consistency monitoring, and multi-objective optimization during complex text generation tasks, providing rare and meaningful insights for the development of future models.

F Models Details

This section provides comprehensive details for the 13 Large Language Models (LLMs) evaluated in our study. To ensure the reproducibility of our experiments, Table 7 lists the specific model versions used. All results from proprietary models were obtained by querying their official APIs.

Our evaluation protocol employs distinct hyperparameter settings depending on the model’s role. For the LLM as an Information Extractor, we used a deterministic setting with a temperature of 0.0. For the tested models performing the open-ended writing tasks, we set the temperature to 0.9. To ensure reliable results against generation variance, each instance was run 3 times, and the average scores are reported.

G Experiment Results

Table 6 also presents a compelling comparison of different prompting strategies across large language models. The results reveal significant variations in how different prompting approaches affect performance. Most strikingly, Few-shot prompting emerges as the consistently superior strategy across all models, with DeepSeek-V3 showing the most dramatic improvement, suggesting that providing multiple demonstration examples substantially enhances the models’ ability to understand and execute complex rewriting constraints. Counterintuitively, Chain of Thought (CoT) prompting which is widely celebrated in other contexts actually diminishes performance across all models, with Claude-3.5-Sonnet suffering the most significant drop, indicating that explicit reasoning processes may interfere with text rewriting capabilities rather than enhance them. The Self Critic approach yields mixed results, benefiting all models but just slightly for ChatGPT-4.1 and Claude-3.5-Sonnet, while One-shot prompting also shows moderate improvements for all three models. These findings highlight the task-specific nature of optimal prompting strategies and suggest that in complex text rewriting scenarios with multiple constraints, concrete examples (particularly multiple examples) provide more actionable guidance than abstract reasoning instructions, with important implications for optimizing LLM performance on sophisticated generation tasks requiring adherence to multiple constraints simultaneously. The universal degradation from coarse to fine-grained constraints across all strategies confirms that current prompting tech-

Model Name	Organization	Version	Access Method
<i>Proprietary Models</i>			
ChatGPT-5.1	OpenAI	gpt-5.1-2025-11-13	Official API
ChatGPT-5.2	OpenAI	gpt-5.2-2025-12-11	Official API
ChatGPT-4o	OpenAI	gpt-4o-2024-08-06	Official API
ChatGPT-3.5-Turbo	OpenAI	gpt-3.5-turbo-0125	Official API
Claude-4.5-Sonnet	Anthropic	claude-sonnet-4-5-20250929	Official API
Claude-4.5-Haiku	Anthropic	claude-haiku-4-5-20251001	Official API
<i>Open-Source Models</i>			
LLaMA-3.1-8B	Meta AI	LLama-3.1-8B-Instruct	Hugging Face
LLaMA-4-Scout	Meta AI	LLaMA-4-Scout	Hugging Face
Qwen3-8B	Alibaba Cloud	qwen3-8b	ModelScope
Qwen3-235B	Alibaba Cloud	qwen3-235b-a22b	ModelScope
DeepSeek-R1	DeepSeek AI	DeepSeek-R1-0120	Hugging Face
DeepSeek-V3.2	DeepSeek AI	DeepSeek-V3.2	Hugging Face

Table 7: Details of the Large Language Models Evaluated in This Study

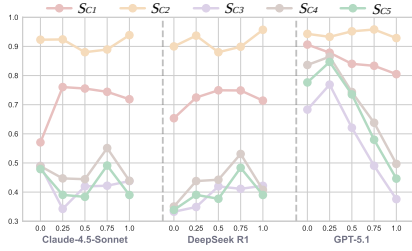


Figure 6: The effect of varying the temperature parameter t . We use $t = 0$ to denote greedy decoding.

niques cannot adequately address the fundamental challenge of maintaining entity-level precision throughout extended generation.

As shown in Figure 6, we conduct an ablation study to assess the impact of decoding temperature t on constraint adherence across all metrics (SC_1 - SC_5) for Claude-4.5-Sonnet, DeepSeek-R1, and ChatGPT-5.1. Claude-4.5-Sonnet maintains stable performance across temperatures, with SC_3 consistently above 0.9. Interestingly, moderate temperature ($t = 0.25$) yields slight gains in SC_1 and SC_5 , suggesting that limited randomness can enhance structural fidelity. DeepSeek-R1 exhibits greater sensitivity: entity-level metrics (SC_4 , SC_5) peak at $t = 0.5$, but performance declines at the extremes. This indicates that mild stochasticity improves surface realization without sacrificing semantic consistency. ChatGPT-5.1 performs best under deterministic decoding ($t = 0$), with steep degradation across all metrics as t increases, par-

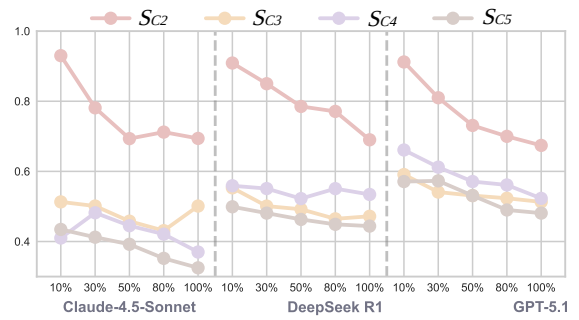


Figure 7: How language models (Claude-4.5-Sonnet, DeepSeek-R1, and GPT-5.1) perform on different evaluation metrics (SC_2 - SC_5) across various positions in generated text.

ticularly beyond $t = 0.5$. This underscores its reliance on low-temperature outputs to preserve the stability of reasoning ability to preserve detailed structure and meaning. Overall, **moderate temperature can benefit some models, but high randomness universally harms fine-grained constraint adherence**. Careful tuning of decoding parameters is thus essential for reliable long-text generation under complex constraints.

The analysis of Figure 7 reveals compelling patterns in how language models (Claude-4.5-Sonnet, DeepSeek-R1, and GPT-5.1) adhere to constraints at different positions within generated long text. Most notably, all three models exhibit a pronounced downward trend in topic coherence (SC_2) as position increases from 10% to 100%, starting

with remarkably high scores (approximately 0.9) at the beginning but declining to approximately 0.65-0.7 by the end, which suggests a significant challenge in maintaining topical relevance throughout extended generations. The relational metrics—sentence relation (S_{C3}), entity type (S_{C4}), and entity relation (S_{C5})—consistently score lower than topic coherence across all positions, generally ranging between 0.4-0.6, and while they show a less dramatic but still consistent downward trend as position increases, the substantial gap between S_{C2} and these other metrics persists throughout the text. Model-specific patterns are also evident: Claude-4.5-Sonnet demonstrates the steepest decline in S_{C2} from early to late positions, DeepSeek-R1 maintains the most stable performance on S_{C3} - S_{C5} across positions, and GPT-5.1 shows slightly better overall S_{C4} performance compared to the other models. These findings highlight a “front-loading” effect where constraint adherence is stronger in early portions of the text and gradually degrades as generation continues—a pattern most pronounced for topic coherence but present across all measured constraints—suggesting a fundamental challenge in long-text generation where maintaining coherence and relationship fidelity becomes increasingly difficult over extended sequences, and pointing to the need for improved techniques that can help models maintain consistent constraint adherence throughout the entirety of generated long text rather than primarily at the beginning.

G.1 Constraint Interaction Analysis

To investigate whether different constraint types interact or interfere with each other as difficulty levels increase from L1 to L4, we conducted a systematic constraint interaction analysis across all 13 evaluated models using paired t -tests, Friedman tests, Cohen’s d effect sizes, and cross-constraint Pearson correlations.

Independence of Coarse-Grained Constraints

The scores for S_{C1} , S_{C2} , and S_{C3} show no statistically significant changes as additional constraints are introduced at successive difficulty levels (all $p > 0.35$, Cohen’s $d < 0.26$). In particular, S_{C1} varies by only 0.21 percentage points across L1–L4, yielding a stability index of 0.86. This confirms that coarse-grained constraint adherence is not degraded by the addition of fine-grained constraints.

Positive Interaction Between Entity-Level Constraints

A notable positive interaction emerges

between S_{C4} and S_{C5} : S_{C4} improves by +2.14 points from L3 to L4 when S_{C5} is introduced ($p = 0.006$, Cohen’s $d = 0.92$). This effect is explained by the extremely high correlation between the two metrics ($r = 0.97$), indicating that they measure closely related entity-level capabilities. The explicit requirement to preserve entity relations (C5) appears to reinforce entity type preservation (C4) as a beneficial side effect.

Correlation Structure Cross-constraint correlation analysis reveals a clear structure: S_{C3} , S_{C4} , and S_{C5} form a tightly correlated capability cluster ($r = 0.83$ – 0.97), while S_{C1} remains largely independent ($r < 0.40$ with S_{C2} , S_{C4} , and S_{C5}). This suggests that fine-grained constraints share underlying generation abilities but do not competitively interfere with coarse-grained ones.

Model-Level Sensitivity We quantified each model’s sensitivity to constraint addition by measuring the total absolute score change ($|\Delta|$) across successive levels. DeepSeek-V3.2 and Claude-4.5-Sonnet are the most stable (total $|\Delta| < 7$), while ChatGPT-5.2 is the most sensitive (total $|\Delta| = 18.0$), indicating that robustness to increasing constraint complexity varies substantially across architectures.

G.2 Robustness to Score Weighting

Since our main results report per-constraint scores alongside an equally weighted average, a natural question is whether model rankings are sensitive to alternative weighting schemes. Equal weighting is the standard convention in constraint-following benchmarks (Jiang et al., 2024), as non-uniform weighting introduces subjectivity with no objective criterion for determining weights.

To verify robustness, we compute an S_{C1} -upweighted average ($w_{C1} = 2.0$, all others $w = 1.0$) to address the concern that topic coherence, as the primary task completion metric, may deserve greater emphasis. The re-ranked results are shown in Table 8.

The rankings remain nearly identical under the alternative weighting, with only a single adjacent swap (Qwen3-8B (thinking) and LLaMA 4 Scout exchange 8th and 9th positions). This confirms that our reported findings are robust to reasonable weighting choices, and that per-constraint scores in Tables 3 and 4 enable readers to re-rank models under any preferred weighting scheme.

Rank	Model	Avg.	Weighted Avg.	Δ
1	Claude-4.5-Sonnet	73.7	77.2	+3.5
2	DeepSeek-V3.2	72.4	76.3	+3.6
3	DeepSeek-R1	70.4	73.4	+3.1
4	Qwen3-235B (thinking)	69.9	73.1	+3.1
5	Qwen3-235B (no thinking)	67.9	72.3	+4.3
6	ChatGPT-5.2	66.9	70.8	+3.9
7	ChatGPT-5.1	64.6	68.7	+4.1
8	Qwen3-8B (thinking)	63.1	68.2	+5.1
9	LLaMA 4 Scout	63.9	67.5	+3.6
10	Qwen3-8B (no thinking)	61.8	66.3	+4.5
11	LLaMA-3.1-8B	60.4	64.2	+3.8
12	ChatGPT-4o	59.0	63.1	+4.1
13	Claude-4.5-Haiku	51.8	56.3	+4.5

Table 8: Model rankings under equal weighting (Avg.) versus S_{C1} -upweighted scoring (Weighted Avg., $w_{C1} = 2.0$). Rankings are nearly identical; the only change is Qwen3-8B (thinking) rising from 9th to 8th, swapping with LLaMA 4 Scout.

We also note that a degenerate identity baseline (copying the input without modification) would achieve high structural constraint scores but would score near zero on S_{C1} , since all entities would belong to the original topic T rather than the target topic T' . The topic coherence constraint thus serves as a natural safeguard against trivial solutions, regardless of its weight.

H Further Analysis

H.1 Validation of Automatic Metrics with Human Judgment

To further validate the effectiveness and reliability of our automatic evaluation metrics, we conducted a human evaluation study. The primary objective was to measure the correlation between the automated scores assigned by *DocFollow* and the judgments of human experts on the fulfillment of the specified constraints. While our benchmark’s main goal is to assess an LLM’s ability to follow fine-grained constraints rather than satisfying subjective preferences, this study provides additional evidence for the reasonableness of our metrics.

Experimental Setup Our human evaluation was structured as follows. We randomly sampled 40 instances from our test set and utilized the outputs generated by four different models to create a pool of pairwise comparisons. Each sampled instance was then routed to the corresponding domain-specific group for evaluation. For each comparison, the designated group of three domain experts was presented with the original input, the five rewrite instructions, and two anonymous, side-by-side model-generated outputs. Their task was

to make a binary choice, selecting the output that better fulfilled all given task requirements.

Agreement and Results The annotation process demonstrated high inter-annotator agreement among the 3-person evaluation groups, achieving a Fleiss’ Kappa score of 0.87, which indicates a strong consensus among the experts. The definitive human preference for each pair was subsequently determined by a majority vote (i.e., at least 2 out of 3) within each expert group. Our analysis revealed a strong alignment between our automated metrics and these human preferences. We define prediction accuracy as the percentage of cases where the model output with a higher automated score was also the one preferred by the human annotators. Across all evaluated pairs, *DocFollow*’s scoring achieved a prediction accuracy of 85%. This result reliably demonstrates that our benchmark effectively captures the core aspects of task success as perceived by domain experts, thus validating its reliability.

Domain-Structured Corpus Collection Guideline
1. Your Goal To curate an authentic, high-quality, and representative corpus for the DocFollow benchmark. 2. Guiding Principles for Domain Experts: Our collection philosophy is built on trust in your professional judgment. Please adhere to the following core principles when selecting texts for your domain. Select for Authenticity, Not Convenience. Choose texts that are genuine, real-world examples of communication in your field. The content should be created by professionals, for professionals. Prioritize a Strong Domain "Signature". The best texts are those that a fellow expert would immediately recognize as belonging to your domain. This means they should be rich in specialized terminology and follow the typical writing style and structure of your field. Avoid generic or overly simplified content. Focus on Core Topics. Select documents that deal with significant, central subjects within your subdomain. The content should be substantive and provide a solid foundation for creating meaningful tasks. Use Reputable, Open-Source Materials Only. All texts must be sourced from publicly accessible, credible platforms (e.g., academic archives, government databases, public-domain libraries). Ensure there are no copyright restrictions for research use. 3. Technical & Submission Requirements: To ensure consistency across the dataset, all submitted texts must meet the following criteria: Source: Must be from a verifiable open-access or public-domain repository. Format: Must be in plain text (.txt) with UTF-8 encoding. Content: Must contain the main body of the document only. All boilerplate (headers, footers, navigation, ads, etc.) must be removed. Length: The word count must be between 600 and 6,500 words. Texts outside this range require specific justification and pre-approval. Metadata: Each text must be submitted with its original source URL.

Figure 8: Dataset Collection Guideline.

Generated Topic Validation Guideline
1. Your Goal To quality control AI-generated topics. For each topic, you will perform a two-step check: Rule Check: Did the AI strictly follow all generation rules (word count, structure, etc.)? Plausibility Check: Is the resulting topic logical and sensible? A topic must pass BOTH checks to be rated "Good". 2. Your Task Judge if the Generated Topic are "Good" (Pass) or "Bad" (Fail). Evaluation Criteria: Content: Is it the most representative summary of the text? Is it a plausible and sensible interpretation? Word Count: Is it strictly 5-15 words? Format: Is it only the topic string (no extra phrases)? Example: Text: An article about the scientific method and validating hypotheses... Good Topic: The importance of empirical evidence in the scientific validation process. Judgment: Good. (Plausible and follows all rules.) Bad Topic (Implausible): The philosophical similarities between scientific hypotheses and musical notes. Judgment: Bad. (Fails Plausibility Check, even if word count is okay.)

Figure 9: Generated Topic Validation Guideline.

Transformed Topic Validation Guideline
1. Your Goal To quality control AI-generated topics. For each topic, you will perform a two-step check: Rule Check: Did the AI strictly follow all generation rules (word count, structure, etc.)? Plausibility Check: Is the resulting topic logical and sensible? A topic must pass BOTH checks to be rated "Good". 2. Your Task Judge if the Original Topic and the Transformed Topic are "Good" (Pass) or "Bad" (Fail). For each stage, first check the structural rules, then perform the final plausibility check. Stage 1: Inversion/Refinement Rules: 1) Replaced only 1-2 words? 2) Used a valid antonym/synonym? 3) Word count & structure are identical? Final Plausibility Check: Is the new meaning sensible? Reject if self-contradictory. Example: Original: The logical progression of a compelling argument. Good Transform: The emotional progression of a compelling argument. Judgment: Good. (Structurally correct and plausible.) Bad Transform: The illogical progression of a compelling argument. Judgment: Bad. (Structurally correct, but fails Plausibility Check as it's self-contradictory.) Stage 2: Parallel Concept Swap Rules: 1) Swapped only one part (P or C)? 2) New concept is from the same domain? 3) Word count & structure are identical? Final Plausibility Check: Is the new combination of Phenomenon and Context logical? Example: Original: The rapid decline of arctic sea ice [P] due to warming temperatures [C]. Good Transform: The rapid decline of coral reefs [P*] due to warming temperatures [C]. Judgment: Good. (Structurally correct and plausible.) Bad Transform: The rapid decline of literacy rates [P*] due to warming temperatures [C]. Judgment: Bad. (Structurally correct, but fails Plausibility Check, the causal link is nonsensical.) Stage 3: Analogous Domain Mapping Rules: 1) New domain is distant but analogous? 2) Preserves logical skeleton? 3) Word count & structure are identical? Final Plausibility Check: Is the mapped analogy meaningful? Reject if absurd. Example: Original: The rapid decline of arctic sea ice due to warming temperatures. Good Transform: The rapid decline of user engagement due to product updates. Judgment: Good. (Structurally correct and the analogy is plausible.) Bad Transform: The rapid decline of moral virtues due to the invention of the fork. Judgment: Bad. (Fails Plausibility Check, the analogy is absurd.)

Figure 10: Transformed Topic Validation Guideline.

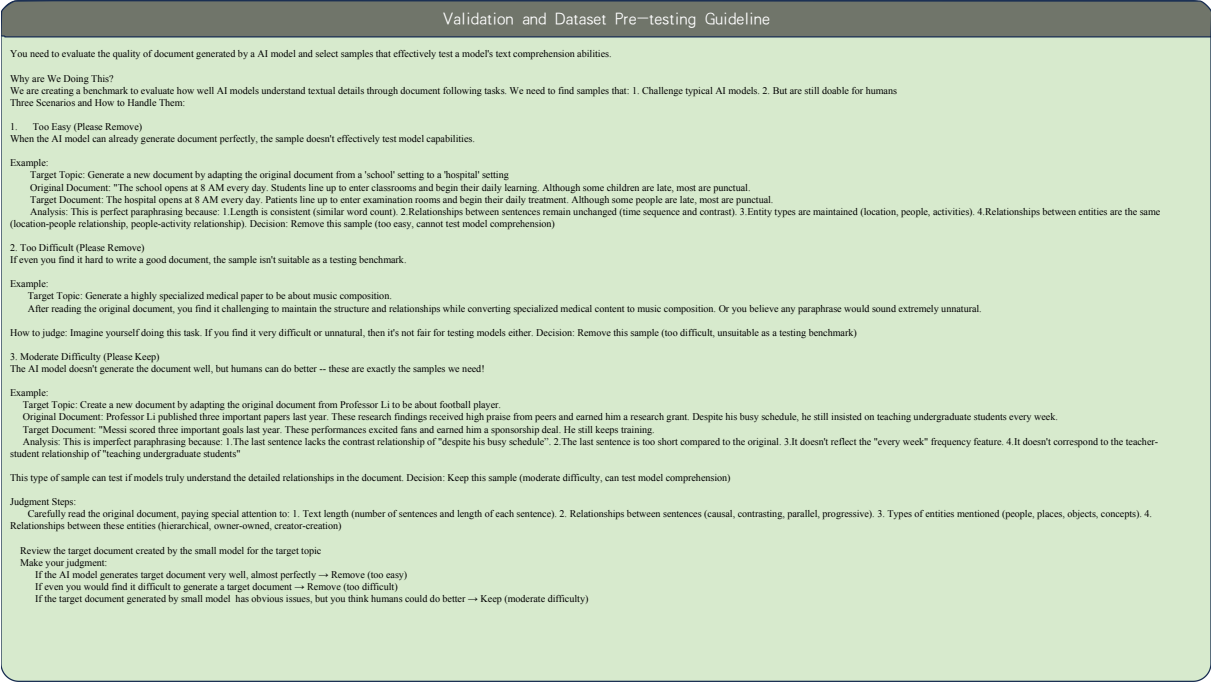


Figure 11: Dataset Pre_testing Guideline.

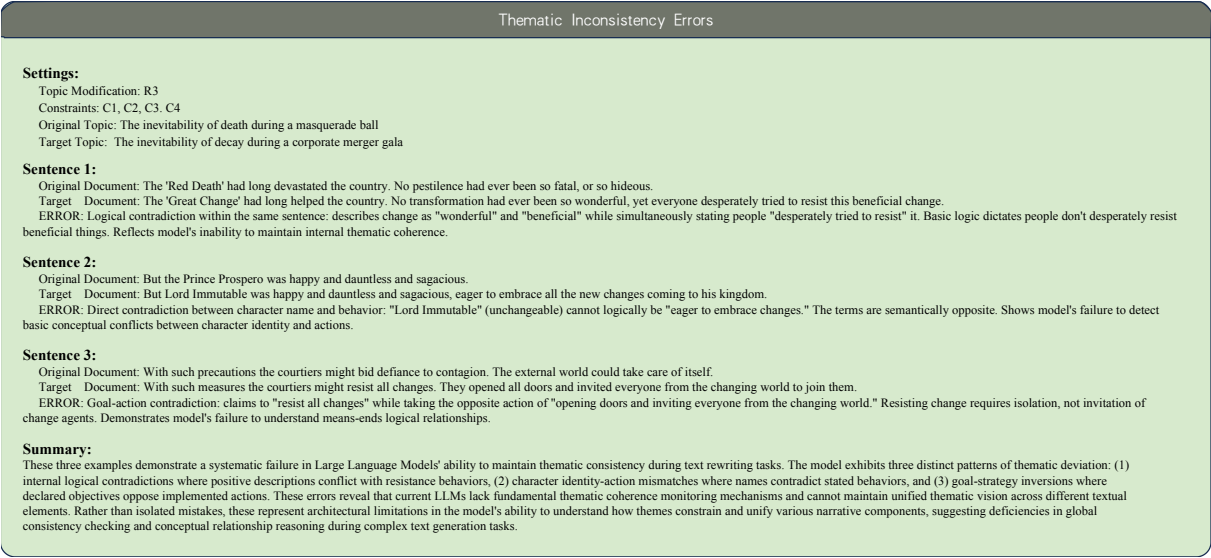


Figure 12: Thematic Inconsistency Errors.

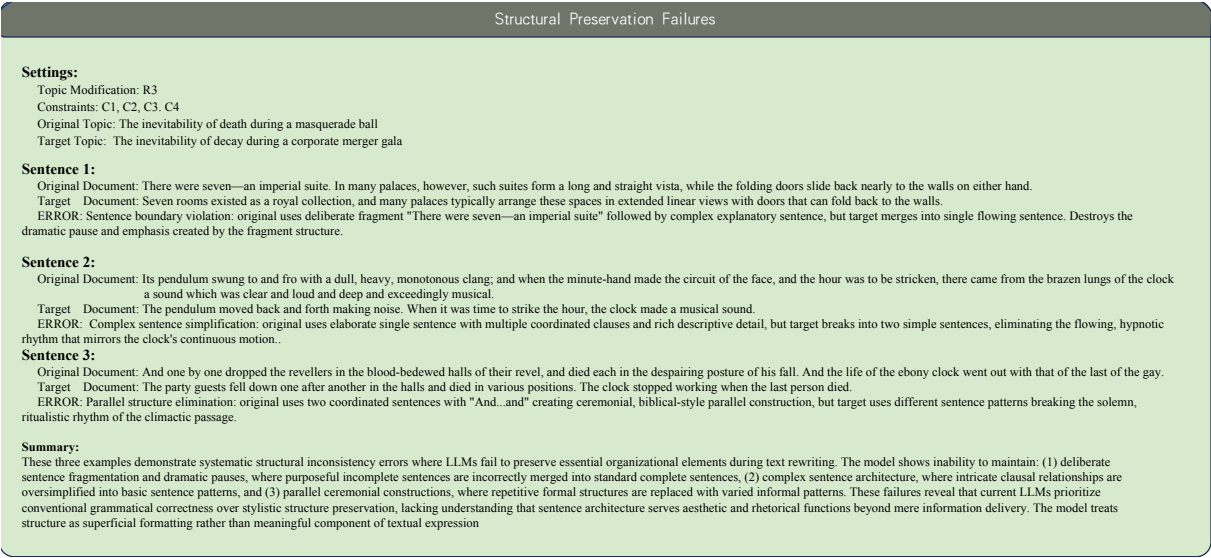


Figure 13: Structural Preservation Failures.

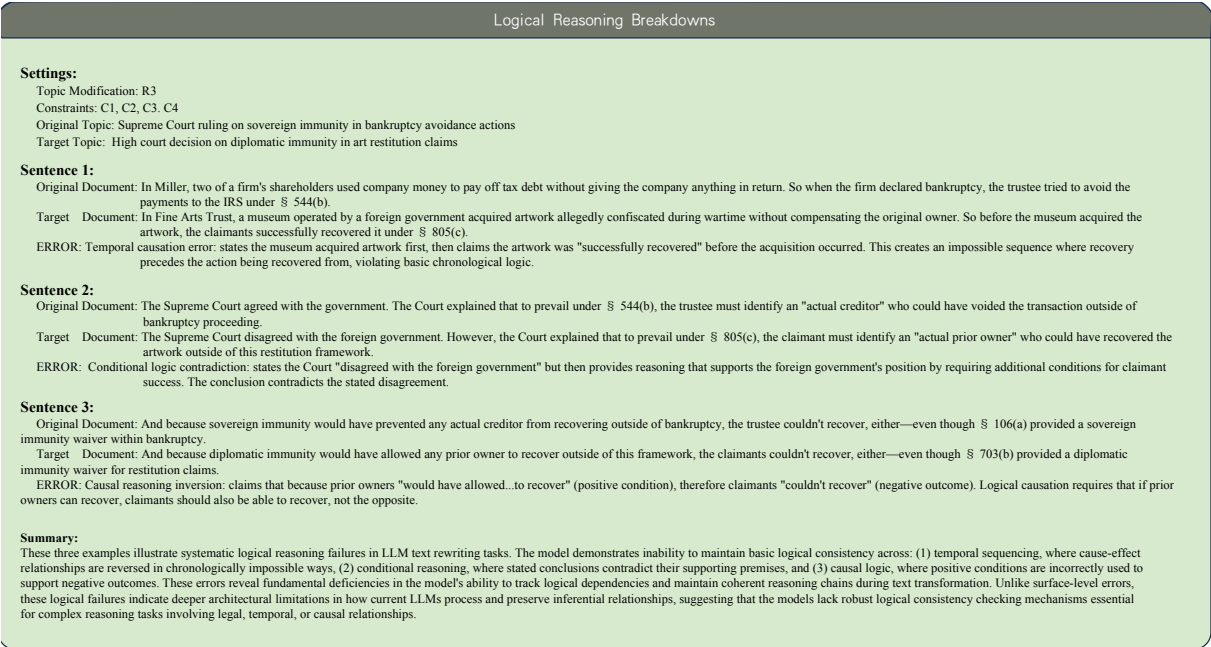


Figure 14: Logical Reasoning Breakdowns.

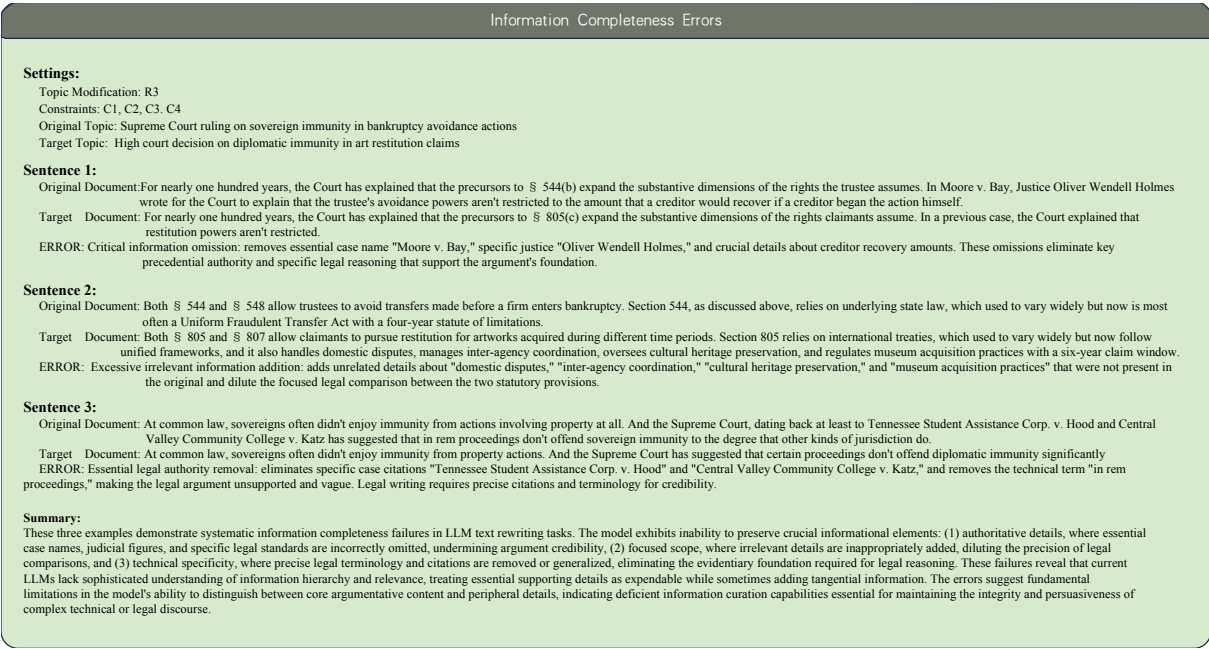


Figure 15: Information Completeness Errors.

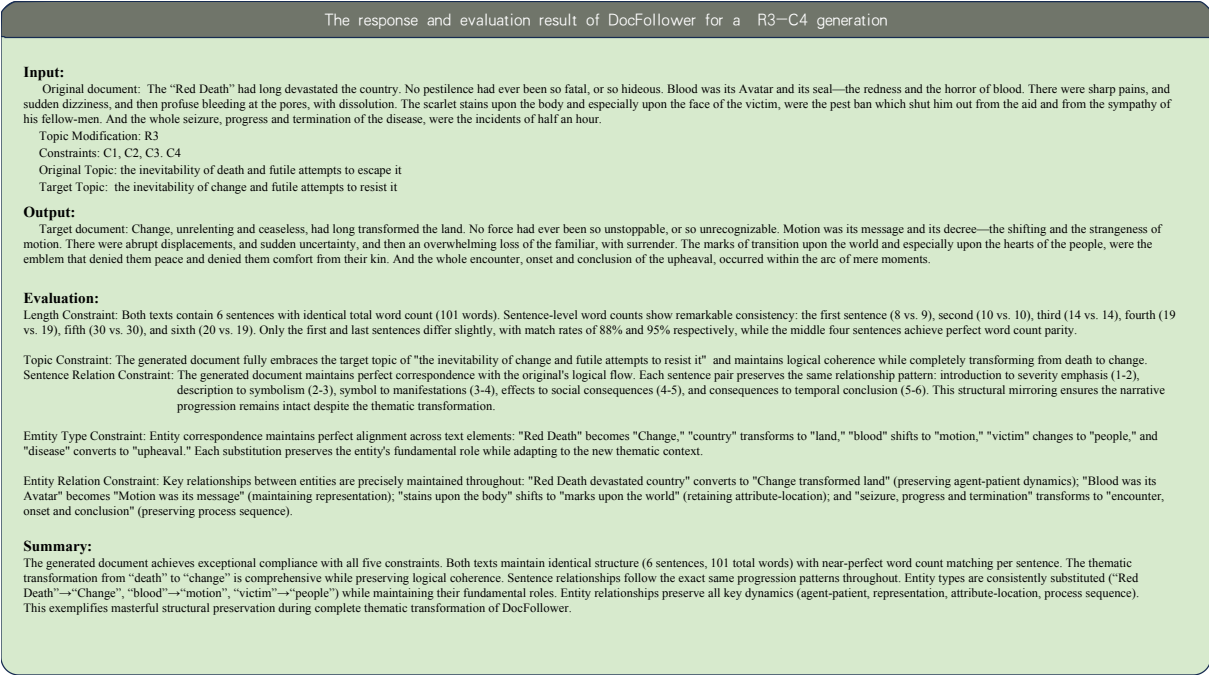


Figure 16: Positive Case Study of *DocFollower* on document following task at the level of R3-C4.

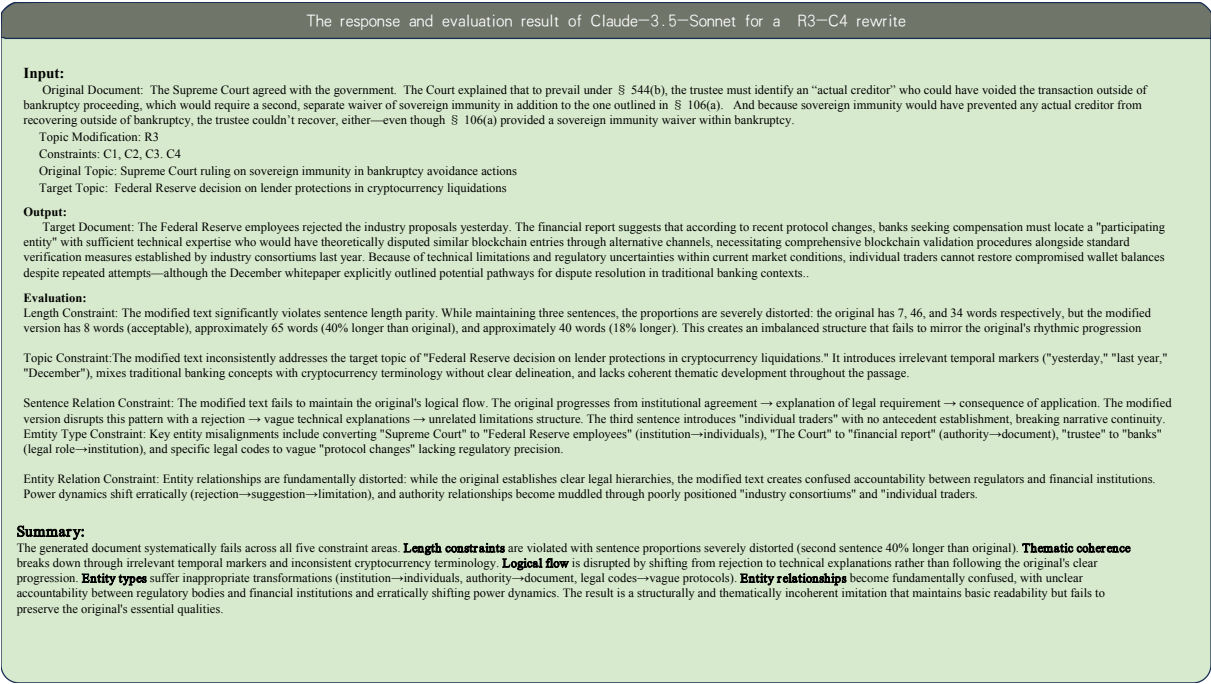


Figure 17: Negative Case Study of Claude-3.5-Sonnet on document following task at the level of R3-C4.

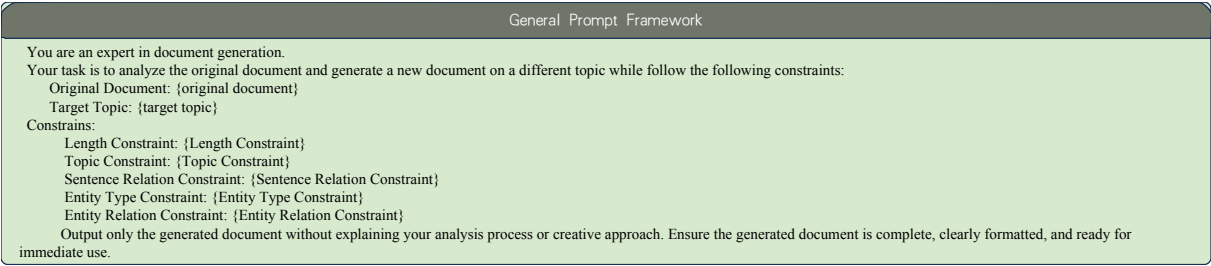


Figure 18: General Prompt Framework.

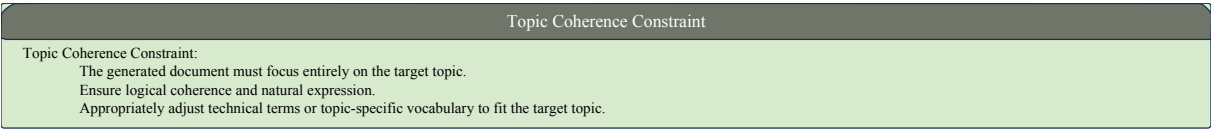


Figure 19: Topic Coherence Constraint.

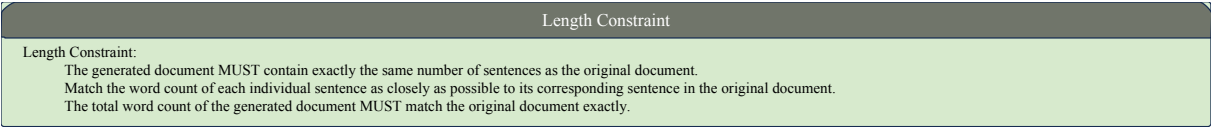


Figure 20: Length Constraint.

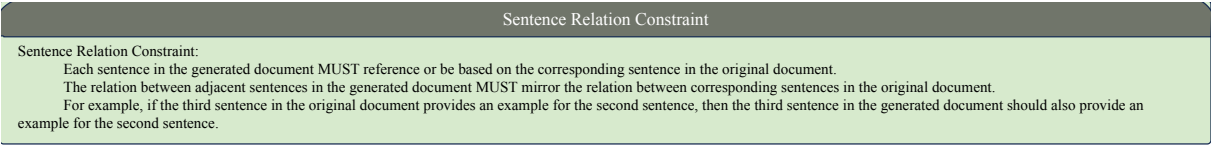


Figure 21: Sentence Relation Constraint.

Entity Type Constraint
Entity Type Constraint: Entities in each sentence of the generated document MUST correspond to entities in the matching sentence of the original document. Ensure that entity types remain consistent (e.g., if the original sentence mentions a person, a location, and an organization, the corresponding sentence should also contain a person, a location, and an organization). Maintain the same pattern and sequence of entities within each sentence when possible.

Figure 22: Entity Type Constraint.

Entity Relation Constraint
Entity Relation Constraint: The relationships between entities in each sentence of the generated document MUST mirror the relationships between entities in the corresponding sentence of the original document. Ensure that relationship types remain consistent (e.g., if entity A is the parent of entity B in the original, then the corresponding entities in the generated document should also have a parent-child relationship). Preserve semantic roles between entities (e.g., if one entity is the agent and another is the recipient in the original, maintain this dynamic in the imitated text)

Figure 23: Entity Relation Constraint.

Get Origin Topic Prompt
Your task is to identify and output the single, most representative main topic from the text provided below. Adhere strictly to these non-negotiable output rules: 1. Your response MUST consist only of the extracted topic itself. DO NOT include any introductory phrases (e.g., "The topic is:", "Here is the topic:", "Okay, the main topic is:"). DO NOT include any explanations, apologies, summaries, or conversational filler. DO NOT include any labels or quotation marks around the topic unless they are part of the topic itself as found in the text. The output must be a direct string of words representing the topic. 2. The topic MUST be between 5 and 15 words in length, inclusive. This word count range is strict. 3. The topic MUST be maximally comprehensive, accurately encapsulating the core subject and primary topics of the text, while strictly adhering to the specified word count. Prioritize full text coverage within the brevity constraint. Text to analyze: {text}

Figure 24: Get Original Topic Prompt.

Topic Modify Prompt
Here is the original topic Q0: "{topic}" Task: Identify the one or two most central content words in Q0 (e.g. key nouns or adjectives). Replace each with either an antonym (to invert) or a near-synonym (to refine) so that the new topic Q1 (a) preserves Q0's structure and rhythm, (b) subtly shifts its meaning, and (c) would still almost—but not quite—fit the original document. (d) has the exact same number of words as Q0. Output only the transformed theme Q1. Do not include any explanations, reasoning, or additional text.""", Here is the original topic Q0: "{topic}" Task: 1. Decompose Q0 into two slots—Phenomenon (P) and Context (C). 2. Choose one of these slots (P or C) and swap it for a parallel concept within the same domain (e.g. replace 'temperature drop' with 'salinity increase' in a marine context). 3. Draft a new topic Q1 that (a) keeps the other slot unchanged, (b) mirrors Q0's syntax exactly, and (c) renders the original document's examples inapplicable without extensive rework. (d) has the exact same number of words as Q0. Output only the transformed topic Q1. Do not provide any explanations or additional text.""", Here is the original topic Q0: "{topic}" Task: 1. Abstract Q0's core argumentative frame (e.g. 'change in X → response in Y'). 2. Select a distant but analogous domain and map X→X* and Y→Y* accordingly. 3. Reproduce Q0's syntactic pattern and rhetorical rhythm to form a new topic Q1 about X*'s effect on Y* in that new context. Ensure Q1 (a) fully breaks topical overlap with the original, yet (b) preserves its logical skeleton for generation. (c) has the exact same number of words as Q0. Output only the transformed topic Q1. Do not include any explanations, steps, or additional text.""",

Figure 25: Topic Modify Prompt.

Predefined Sentence Relation
"Causality": "B expresses a causal link with A, either giving a reason (cause) or stating a result (effect). Keywords: because, therefore, leads to, results in.", "Condition": "B specifies a prerequisite or necessary condition for A to occur or hold. Keywords: if, unless, provided that.", "Purpose": "B states the intended goal or purpose underlying A. Keywords: to, in order to, so that.", "Comparison": "B draws a similarity or contrast between A and another entity or idea. Keywords: like, unlike, similarly, whereas, compared to.", "Expansion": "B provides additional context, background, or elaboration on A without changing its core meaning.", "Rewriting": "B reformulates A using different wording, summarizing or restating its semantic content.", "Summary": "B provides a concise overview or generalization of A's main content. Keywords: in summary, to sum up, overall.", "Opinion": "B offers a subjective judgment, evaluation, interpretation, or recommendation about A.", "Sequence": "B indicates the chronological or logical order relating to A, marking progression. Keywords: first, next, then, subsequent.", "Transition": "B signals a shift to a new topic or perspective without elaborating on A. Keywords: now, moving on, regarding.", "Definition": "B defines or clarifies the meaning of a term or concept from A. Keywords: is defined as, means, refers to.", "Question": "B poses a question or inquiry about A. Keywords: why, what, how, does, is?", "Exception": "B introduces an exception or limitation to the assertion in A. Keywords: except that, apart from, aside from.", "Others": "B relates to A in a way that doesn't clearly fit any of the other defined relationship categories. This is a catch-all category for relationships that are ambiguous or unique."

Figure 26: Predefined Sentence Relation.

Predefined Entity Type
<pre>COMMON_ENTITY_TYPES = ["Person: specific individuals such as scientists, business leaders, educators, artists, or politicians." "Organization: institutional entities, including companies, universities, schools, museums, and government bodies." "Location: geographic places, such as countries, cities, regions, and natural features." "Facility: man-made structures or venues, such as airports, buildings, and landmarks." "Event: historical or planned occurrences, such as conferences, festivals, or historical incidents." "Date/Time: specific dates, time periods, or historical eras relevant to any domain." "Quantity/Measure: measurements with physical units applicable to any field." "Number/Identifier: unitless numbers, codes, or identifiers used across domains." "Concept/Object: other concrete objects or abstract concepts not covered elsewhere." "Others: entities that do not clearly fit into any other defined category. This is a catch-all category for ambiguous or unique entities."]</pre> <pre>ACADEMIC_ENGINEERING_ENTITY_TYPES = ["Technology: specific technological innovations, systems, or methodologies such as quantum computing, blockchain, or AI algorithms; includes both established and emerging technologies." "Scientific Method: specific research methodologies, experimental designs, analytical approaches, or systematic procedures used in scientific inquiry; includes methodological frameworks unique to different scientific disciplines." "Theory: scientific theories, principles, laws, hypotheses, or academic frameworks that explain phenomena or guide research; includes both established and emerging theoretical constructs." "Academic Institution: research centers, laboratories, academic departments, or specialized scientific facilities dedicated to advancing knowledge in specific disciplines; distinct from general educational organizations." "Innovation Outcome: patents, scientific discoveries, research findings, technological breakthroughs, or engineering solutions that represent significant contributions to their fields; includes both tangible and conceptual innovations."]</pre>

Figure 27: Predefined Entity Type.

Predefine Entity Relation
<pre>GENERAL_ENTITY_RELATION_TYPES = ["Structural": "Entity A is structurally related to Entity B through composition, containment, or organization. Example: [Chapter] <-> [Book] (part-whole), [Department] <-> [Organization] (unit-organization), [Member] <-> [Committee] (element-collective). Focuses on inherent structural relationships rather than transient locations." "Spatial": "Entity A is physically or geographically related to Entity B. Example: [Building] <-> [City] (located in), [Mountain] <-> [Valley] (adjacent to), [Star] <-> [Galaxy] (situated within). Focuses on physical positioning rather than administrative boundaries." "Temporal": "Entity A relates to Entity B through time sequences, periods, or temporal associations. Example: [Ancient Period] <-> [Modern Era] (precedes), [Event] <-> [Date] (occurs on), [Process] <-> [Duration] (spans). Focuses purely on temporal aspects rather than causal relationships." "Functional": "Entity A serves a purpose, function, or utility for Entity B. Example: [Tool] <-> [Task] (facilitates), [Medicine] <-> [Disease] (treats), [Software] <-> [Business Process] (automates). Focuses on utility relationships independent of causal or transformative aspects." "Causal": "Entity A directly causes, produces, prevents, or impacts Entity B. Example: [Rain] <-> [Flood] (causes), [Vaccine] <-> [Disease] (prevents), [Policy] <-> [Outcome] (influences). Focuses on direct causal linkages rather than enablement or correlation." "Derivational": "Entity A is transformed into, derived from, or serves as raw material for Entity B. Example: [Data] <-> [Information] (processed into), [Draft] <-> [Final Document] (develops into), [Source Material] <-> [Product] (transformed into). Focuses on transformation processes rather than simple causation." "Attributive": "Entity A possesses Entity B as a property, characteristic, or measurable attribute. Example: [Product] <-> [Quality] (has property), [Person] <-> [Age] (has attribute), [Material] <-> [Strength] (has characteristic). Focuses on intrinsic properties rather than subjective assessments." "Hierarchical": "Entity A is related to Entity B through classificatory, taxonomic, or power hierarchies. Example: [Manager] <-> [Employee] (supervises), [Species] <-> [Genus] (belongs to classification), [Subclass] <-> [Class] (is type of). Focuses on vertical organizational relationships rather than horizontal structural ones." "Associative": "Entity A is linked to Entity B through social, contractual, or institutional arrangements. Example: [Person] <-> [Organization] (is member of), [Party A] <-> [Party B] (has agreement with), [Student] <-> [University] (is enrolled in). Focuses on formalized associations rather than inherent structural or hierarchical relationships." "Comparative": "Entity A is compared to Entity B through similarity, difference, or relative measurement. Example: [Alternative A] <-> [Alternative B] (contrasts with), [Benchmark] <-> [Sample] (serves as standard for), [Original] <-> [Copy] (is replicated by). Focuses on explicit comparison rather than mere co-occurrence." "Others": "Entity A relates to Entity B in a way that doesn't clearly fit any of the other defined relationship categories. This is a catch-all category for relationships that are ambiguous, complex, or unique and cannot be accurately classified using the existing taxonomy."]</pre> <pre>ACADEMIC_ENGINEERING_RELATION_TYPES = ["Epistemic": "Knowledge relationship where Entity A provides evidence for, contradicts, or advances understanding of Entity B. Example: [Experiment] <-> [Theory] (supports/refutes), [Research] <-> [Knowledge Gap] (addresses). Focuses on evidential relationships in knowledge production." "Innovation-Application": "Entity A represents a novel solution or approach that is applied to Entity B. Example: [Algorithm] <-> [Problem Domain] (solution-challenge), [Technology] <-> [Industry] (innovation-implementation context). Focuses on novel implementations rather than standard applications." "Specification": "Entity A defines requirements, parameters, or standards for Entity B. Example: [Technical Standard] <-> [Implementation] (defines-conforms to), [Requirements Document] <-> [Design] (specifies-fulfills). Focuses on normative guidelines rather than descriptive documentation." "Expertise-Field": "Entity A possesses specialized knowledge or capability within domain B. Example: [Specialist] <-> [Discipline] (expert in-domain of expertise), [Research Group] <-> [Research Area] (specializes in-specialty area). Focuses on knowledge competency rather than general associations." "Methodological": "Entity A provides a systematic approach for studying, analyzing, or developing Entity B. Example: [Scientific Method] <-> [Phenomenon] (investigative approach-subject of study), [Development Framework] <-> [System] (methodology-product). Focuses on procedural frameworks rather than specific techniques."]</pre>

Figure 28: Predefined Entity Relation.

Topic Coherence Prompt

Please evaluate this sentence based on the following three criteria: relevance, appropriateness, and coherence with context.

Topic: {topic}
Sentence: {sentence}
Surrounding Context: {context}

For each criterion, determine if the sentence meets the standard (Yes/No), provide a brief explanation, and list any inappropriate words or phrases that cause issues.

Relevance: Is the sentence directly related to the 'Topic' or intended subtopic?
Appropriateness: Is the sentence's language, style, and tone suitable for the given 'Topic' and consistent with the 'Surrounding Context'? Does it avoid jarring shifts in formality, vocabulary, or tone compared to the surrounding text?
Coherence: Does the sentence logically follow from or connect to the 'Surrounding Context' while maintaining consistency with the 'Topic'?

Respond in JSON format:

```

{{
  "relevance": {{
    "assessment": "Yes/No",
    "explanation": "Brief reason",
    "inappropriate_words": ["word1", "word2", ...]
  }},
  "appropriateness": {{
    "assessment": "Yes/No",
    "explanation": "Brief reason",
    "inappropriate_words": ["word1", "word2", ...]
  }},
  "coherence": {{
    "assessment": "Yes/No",
    "explanation": "Brief reason",
    "inappropriate_words": ["word1", "word2", ...]
  }}
}}
```

Figure 29: Topic Coherence Prompt.

Sentence Relation Extraction Prompt

You are an expert linguist specialized in discourse analysis. Your task is to identify the primary relationship between two consecutive sentences based on a predefined list of functions. Follow these instructions precisely:

1. Analyze Function: Determine the main communicative function of Sentence 2 in relation to Sentence 1.
2. Use Definitions: Base your classification strictly on the relationship definitions provided below.
3. Select Exactly One: Choose the single best relationship type from the list that accurately describes this primary function.
4. Verbatim Response: Your response must consist ONLY of the chosen relationship name, exactly as it appears in the list. Do not include explanations, justifications, or any other text.
5. Others' Category: Only select the "Others" category if none of the other relationship types reasonably apply. Exhaust all other possibilities before choosing "Others".

Available Relationship Types and Definitions:
{RELATION_DEFINITIONS}

Classification Task:
Carefully analyze the relationship between Sentence 1 and Sentence 2. Choose ONLY ONE relationship from the following list: {"", ".join(RELATION_DEFINITIONS)}

Selection Process:

- Evaluate Sentence 2's primary function relative to Sentence 1 against the provided definitions.
- Try to match with one of the specific relationship types (Causality, Condition, Purpose, etc.) first.
- Select "Others" ONLY if all other relationship types have been considered and none of them fit the relationship.

Always prefer a more specific category over the "Others" category when possible.

Input Sentences:
Sentence 1: {sent1}
Sentence 2: {sent2}

Relationship: " "

Figure 30: Sentence Relation Extraction.

Sentence Matching Prompt

Task: Find the corresponding sentence from candidates for the given source sentence.

Source sentence: {target sent}

Candidate sentences: {candidate text}

Matching Criteria:
Compare the source sentence with each candidate, select the one that best matches the source sentence.

Important Note:
The topics may be different, focus on how information is presented rather than semantic similarity.

Output Instructions:

- If a corresponding sentence is found: return ONLY its index number (0-based)
- If no corresponding sentence exists: return ONLY "No-Match"
- Do not include any explanations or additional text

Response format:

- Match found: "0" or "1" or "2" etc.
- No match: "No-Match"

Figure 31: Sentence Matching Prompt.



Figure 32: Entity Extraction Prompt.

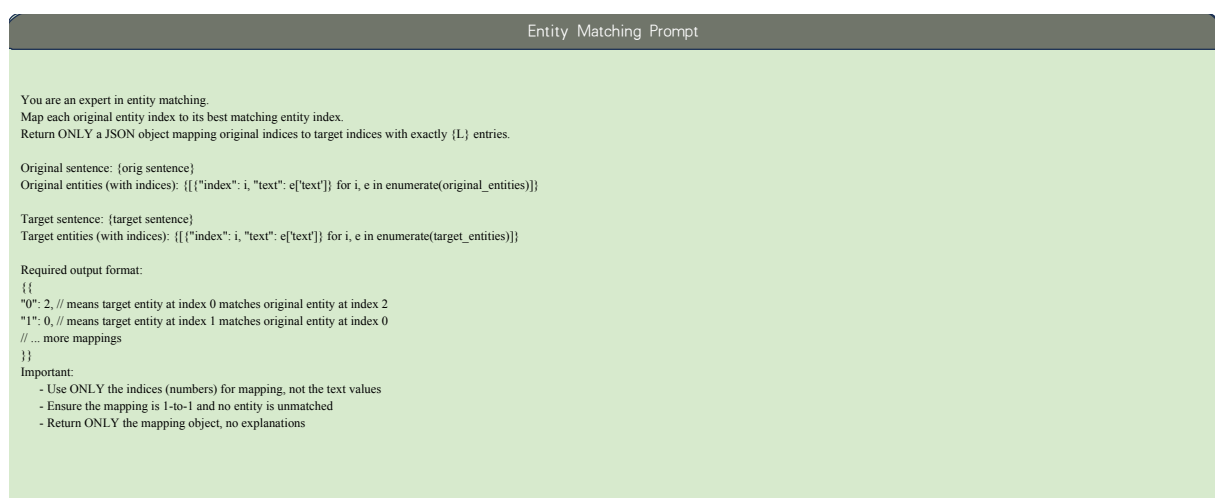


Figure 33: Entity Matching Prompt.

Entity Relation Extraction Prompt

Objective: Extract and output entity relations from TWO input sentences. You MUST analyze BOTH sentences and output their relations in a single response. Your output MUST contain relations for BOTH sentences, not just one.

Predefined Relation Taxonomy and Definitions:
{Formatted_Relation_Definitions}

=== SENTENCE PAIR TO ANALYZE ===
First Sentence: "{sent1}"
First Sentence Entities: {actual_orig_entities}

Second Sentence: "{sent2}"
Second Sentence Entities: {actual_rewrite_entities}

Processing Instructions:

1. Independent Analysis: Process Sentence 1 (using its text and its Detected Entities list) and Sentence 2 (using its text and its Detected Entities list) separately. You MUST analyze BOTH sentences and provide relations for each.
2. Intra-Sentence Relations Only: Identify relationships ONLY between pairs of entities listed in the Detected Entities for the SAME sentence. Do not establish relations between entities from different sentences.
3. Candidate Identification: For each sentence, consider potential relationships between all valid pairs of entities provided for that sentence (e.g., Entity A <-> Entity B).
4. Relation Classification: Classify identified relationships using the types defined in the 'Predefined Relation Taxonomy and Definitions' section above. Carefully consider the definition of each relation type. The classification should be based on the most evident and fitting relation according to the definition and sentence context.
5. Strict Rule Adherence: Apply validation rules meticulously:
 - Entity Matching: The 'subject' and 'object' values in the output relation MUST exactly match the 'text' field of entities present in the corresponding input 'Detected Entities' list.
 - Type Constraints: Ensure the nature of the subject and object entities aligns with the definition of the chosen relation type.
 - Others Category: Use the "Others" category ONLY as a last resort when none of the other predefined relation types clearly fit the relationship between entities. You must exhaustively consider all other relation types before classifying a relationship as "Others".

All Relationships Required: All detected entity pairs that have a relationship should have a relation output, even if the relation type is "Others".

FINAL CHECK:
- Have you analyzed BOTH sentences?
- Does your output contain EXACTLY TWO objects in the result array?
- Are all entity texts exact matches?
- Have you considered all possible entity pairs?
- Have you applied the same analysis depth to both sentences?
- Is your output wrapped in a JSON object with a "result" field containing the array?

Now, analyze the provided sentence pair and output their relations following this format exactly.

Figure 34: Entity Relation Extraction Prompt.

Entity Relation Matching Prompt

You are an expert in relation matching.
Map each original relation index to its best matching target relation index.
Return ONLY a JSON object mapping original indices to target indices with exactly {L} entries.

Original sentence: {original sentence}
Original relation triples with indices: {{{"index": i, "triple": triple} for i, triple in enumerate(triple_strings_original)}}

Target sentence: {target sentence}
Target relation triples with indices: {{{"index": i, "triple": triple} for i, triple in enumerate(triple_strings_target)}}

Required output format:

```
{{
  "0": 2, // means target relation at index 0 matches original relation at index 2
  "1": 0, // means target relation at index 1 matches original relation at index 0
  // ... more mappings up to {L} entries total
}}
```

Important rules:

1. Output ONLY the index mapping JSON object
2. Each original index must map to exactly one target index
3. Use ONLY numerical indices (0, 1, 2, etc.)
4. Include exactly {L} mappings
5. Ensure all indices are valid (within range of the provided triples)
6. If both relations are classified as "Others", consider it a valid match too - this means relationships where both original and target use "Others" classification should be counted as matching

Figure 35: Entity Relation Matching Prompt.

COT Prompt

Chain of Thought Process:

Before generating the final target document, work through the following steps:

1. Count the total number of sentences in the original document and note the word count of each sentence.
2. Identify the key entities in each original sentence and their relationships.
3. Analyze how each sentence connects to adjacent sentences (providing examples, elaborating, contrasting, etc.).
4. For each original sentence, draft a corresponding new sentence on the target topic that:
 - Maintains similar word count
 - Contains matching entity types in the same sequence
 - Preserves the same entity relationships
 - Connects to adjacent sentences in the same way as the original document
5. Verify that your target document meets all five constraints by comparing it sentence-by-sentence with the original document.
6. Make final adjustments to ensure exact total word count and the structure are the same as the original document.

Output Requirements:

Output only the generated document without explaining your analysis process or creative approach. Ensure the output text is complete, clearly formatted, and ready for immediate use.

Figure 36: COT Prompt.

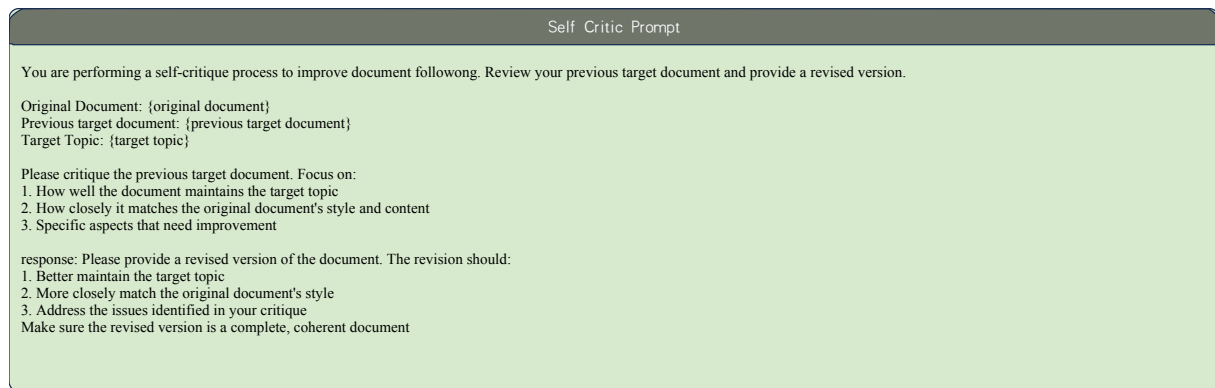


Figure 37: Self Critic Prompt.

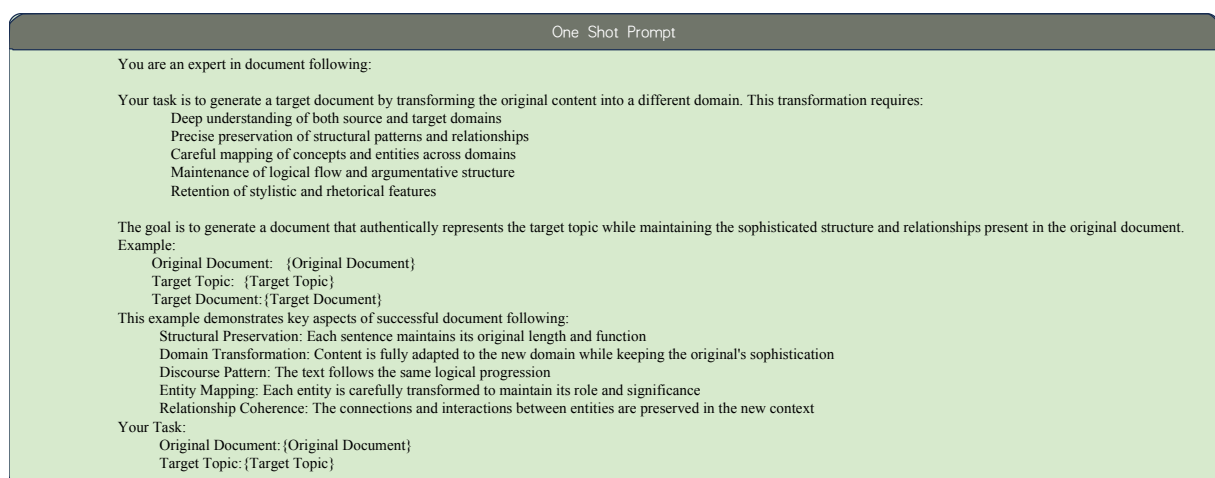


Figure 38: One Shot Prompt.

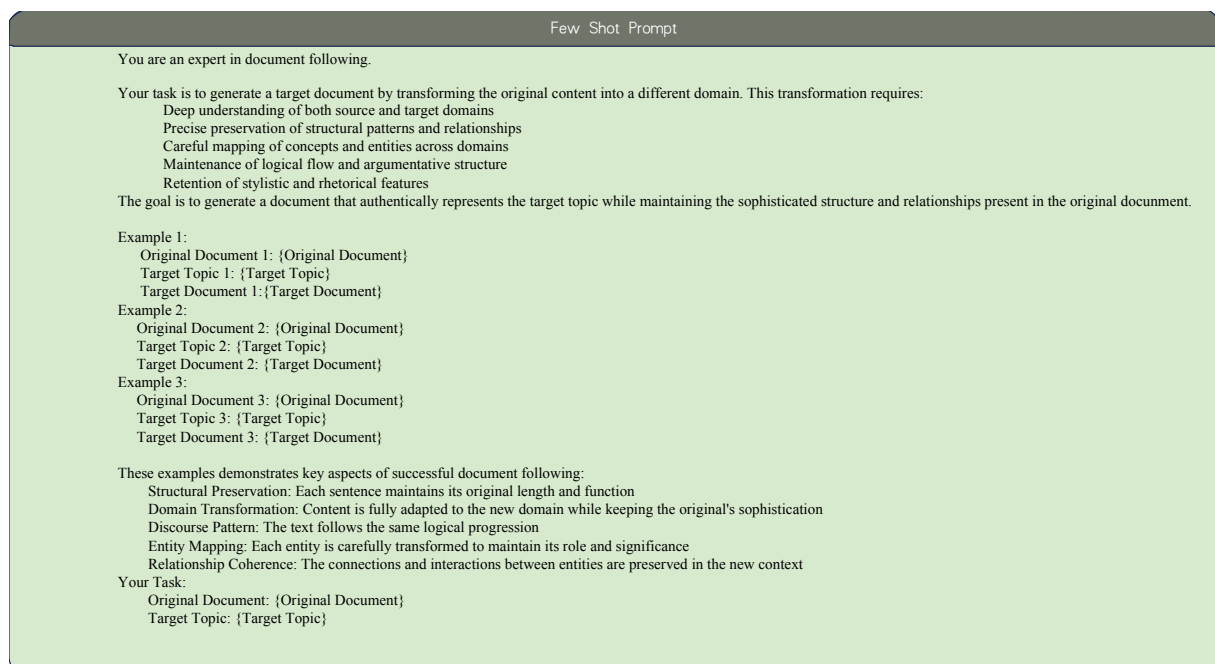


Figure 39: Few Shot Prompt.

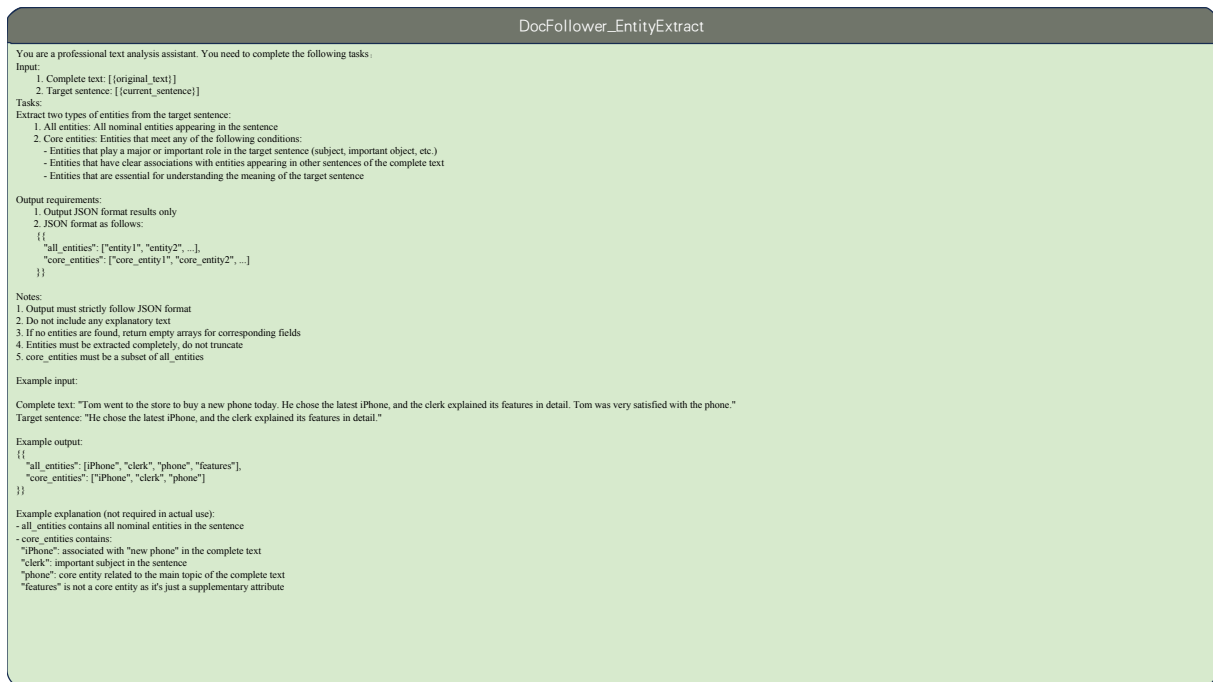


Figure 40: *DocFollower_EntityExtract*.



Figure 41: *DocFollower_RelationExtract* Part 1.

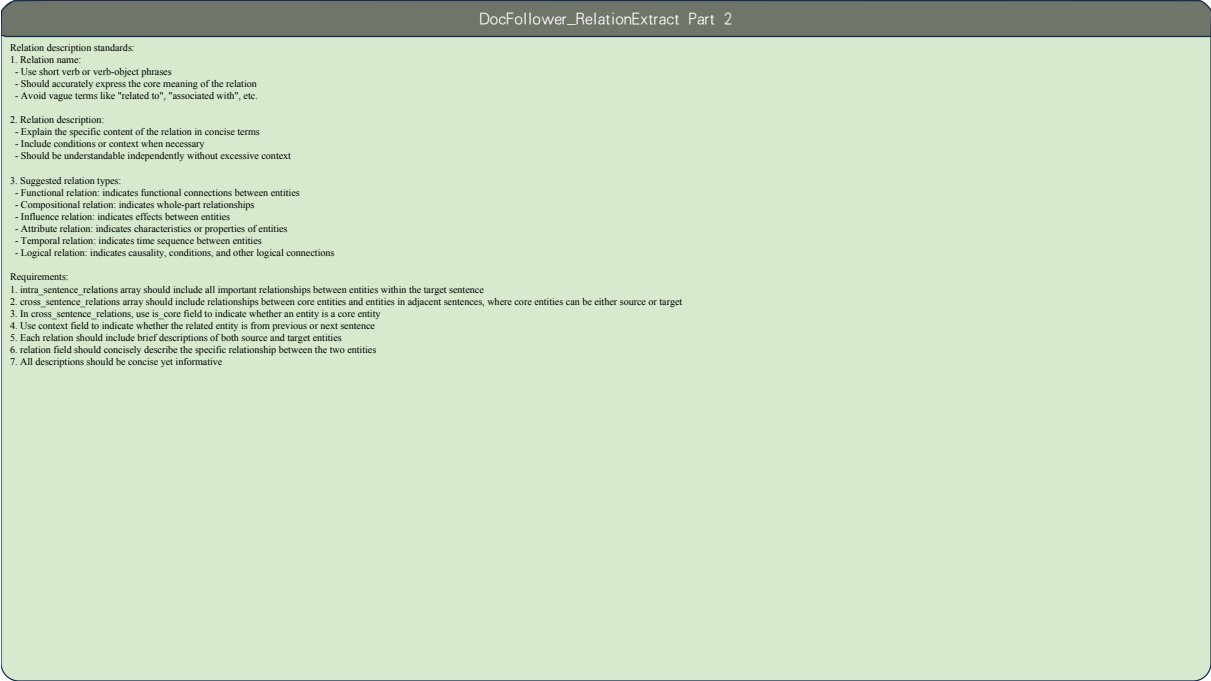


Figure 42: *DocFollower_RelationExtract* Part 2.

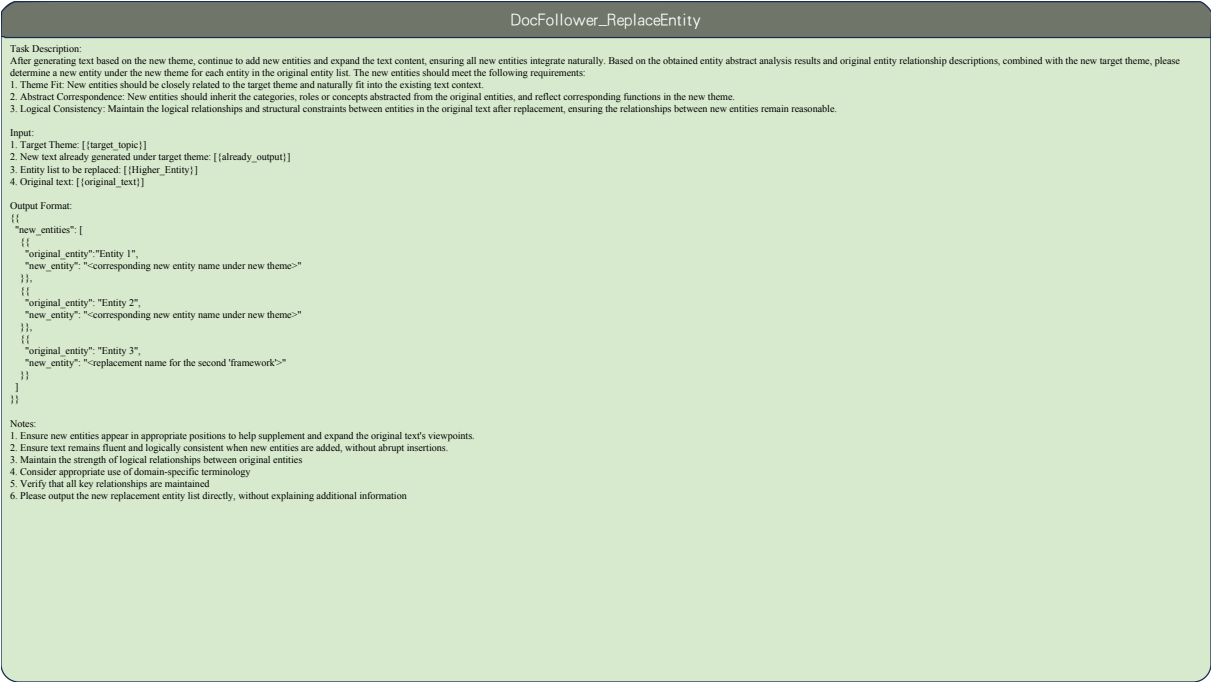


Figure 43: *DocFollower_ReplaceEntity*.



Figure 44: *DocFollower_HighEntityExtract* Part 1.



Figure 45: *DocFollower_HighEntityExtract* Part 2.



Figure 46: *DocFollower_HighRelationExtract*.

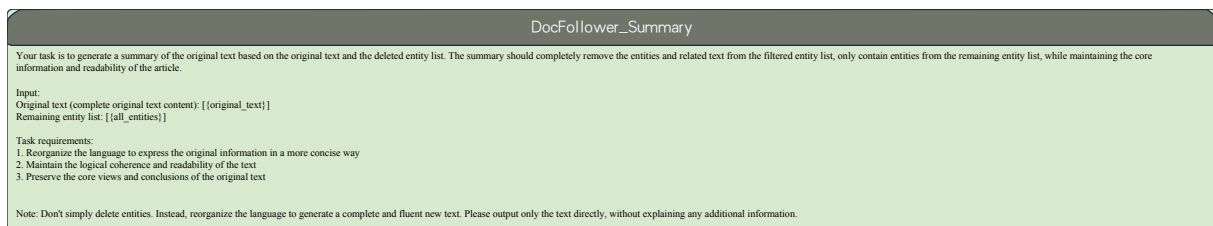


Figure 47: *DocFollower_Summary*.

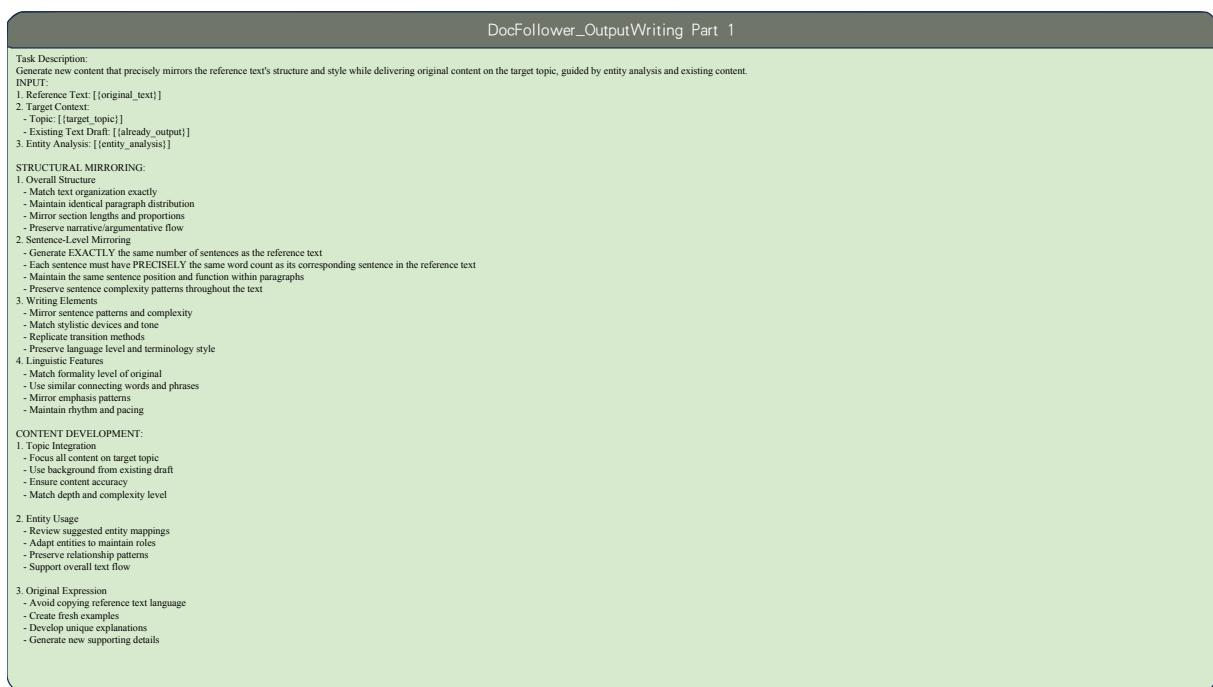


Figure 48: *DocFollower_OutputWriting Part 1*.

DocFollower_OutputWriting Part 2
QUALITY STANDARDS: - Structure Match - Organization mirrors original - Paragraph functions align - Flow patterns match - Complexity level maintained - Content Quality: (1)Topic relevance (2)Factual accuracy (3)Logical flow (4)Appropriate support - Style Consistency: (1) Matching tone (2) Similar language level (3) Equivalent formality (4) Genre-appropriate expression ENTITY INTEGRATION: - Analysis Application - Evaluate each mapped entity - Check contextual fit - Consider alternatives if needed - Verify relationship patterns - Adaptation Process - Adjust for target context - Maintain functional roles - Ensure natural flow - Preserve relationships - Quality Check - Verify entity functions - Check relationship logic - Ensure flow support - Match depth level OUTPUT REQUIREMENTS: - Content - Complete, polished text - Topic-focused - Original expression - Matching complexity - Structure - Exact organization match - Parallel development - Consistent flow - Effective transitions - Style - Matching tone - Similar language level - Equivalent formality - Genre-appropriate expression Final Verification: - Confirm exact structural match - Verify content originality - Check style consistency - Validate entity integration - Ensure topic focus - Verify EXACT sentence count match with reference text - Confirm EACH sentence has the same word count as its corresponding reference sentence Requirement: Output complete text directly, without any additional content.

Figure 49: *DocFollower_OutputWriting Part 2.*