Measuring Information Distortion in Hierarchical Ultra long Novel Reconstruction: The Optimal Expansion Ratio

Anonymous ACL submission

Abstract

A two-stage novel generation framework (outline -> section outline -> manuscript) is widely used in long novel generation,(e.g., DOME, PLAN&WRITE, LONG WRITER), but study of such framework in ultra long novel(>1M words) reconstruction is little.

Building on recent text compression methods (LLMZIP, LLM2VEC), we conduct an information-theoretic analysis to quantify semantic distortion under different compression–expansion ratios. We examine how outline length affects information preservation.

Experiments on ultra-long novels show that the optimal compression-expansion ratio significantly reduces semantic distortion compared to other non-optimal compression-expansion ratio.

1 Introduction

We observe a fundamental phenomenon in large-scale novel generation (Li et al., 2022; Su et al., 2022): When a million-word novel is summarized into a 1,000-word outline by an LLM (e.g., ChatGPT) and then expanded, the result shows substantial distortion, detail loss, and semantic divergence. In contrast, compressing a 100k-word novel into a 10k-word outline and expanding it preserves meaning more faithfully.

Studies such as LongWriter agree that generating 100k+ word novels typically follows an outline-to-novel workflow. The quality depends on two prompt-side variables: (i) outline length, and (ii) detail density (e.g., characters, plots, scenes). A trade-off emerges: fewer tokens and sparse detail reduce quality, while rich, lengthy prompts shift the burden to the human author.

Our ultimate goal is to generate detail-rich ultra-long novels from minimal outlines. However, since "detail" itself is hard to define and evaluate directly, we instead study the detail loss in reconstruction of ultra-long novels under varying compression ratios using large language models, as a proxy for generation quality.

1.1 Motivation

Ultra-long novels (1M+ words) are highly popular on *WuxiaWorld*, *Fanqie*, and *Qidian*, making automated generation a key goal. Since ChatGPT, models like LLaMA (Grattafiori et al., 2024), DeepSeek (DeepSeek-AI et al., 2025), Qwen (Bai et al., 2023), Gemini (Team et al., 2024), and GPT-4o (OpenAI et al., 2024) have advanced long-context processing.

However, despite 1M-token input capacity, output limits (e.g., 16k) make faithful reconstruction difficult (Mikhaylovskiy, 2023). Although many prior studies have addressed novels up to 100k words, empirical evidence is lacking to show that methods effective at this scale naturally extend to generating novels of 1 million words. The challenges on the million-word scale, maintaining coherence, thematic consistency, and character development, are qualitatively and quantitatively different. Inspired by the encoder–decoder paradigm, we adopt a reconstruction-based framework as a surrogate objective to study and improve ultra-long text generation.

Model	Context Size	Max TPM	Max Output
Gemini 2.0 Flash	1,000,000	1,000,000	8,192
Claude 3.7 Sonnet	200,000	200,000	128,000
GPT-4.1	1,000,000	400,000	16,000
Chatgpt-4o	128,000	800,000	16,384
LLAMA 4 Scout	10,000,000	1,000,000	8,000
OPENAI O3	200,000	200,000	100,000

Table 1: Comparison of large language models by context size, maximum tokens processed in a minute, and maximum output length in tokens. On average, one token corresponds to roughly 3 words in English, 2 in Chinese. We quote the tier 3 TPM limit for GPT-4.1, OPENAI O3, and tier 3 TPM limit for Claude 3.7 Sonnet. Data are collected from their websites.

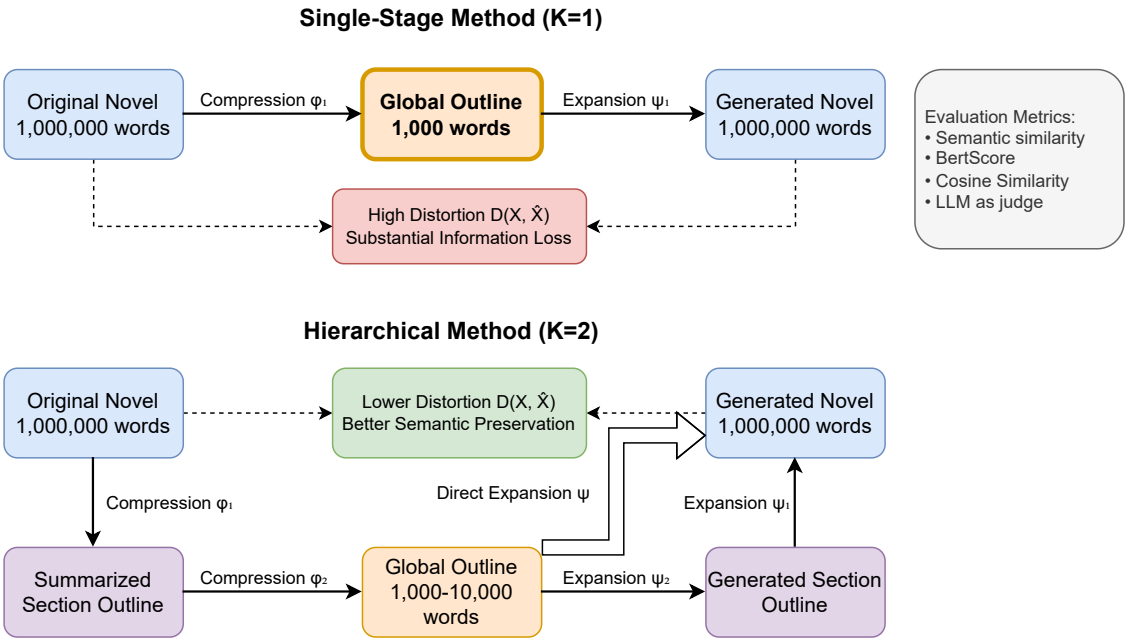

Figure 1: Pipeline for ultra-long novel generation using hierarchical outline approach. Our method maintains narrative fidelity by optimizing compression-expansion ratios across multiple stages.

1.2 Contribution

- We establish a quantitative relationship between outline compression-expansion ratios and distortion (detail loss) through experiments. $R = 0.01$ is the optimal compression-expansion ratio under our configuration and experiment.
- We propose and empirically verify that mixed two-stage hierarchical outline decreases distortion in generated novels compared to traditional one-stage and two-stage approaches.

1.3 Related Work

Long-text Generation. Long-text generation introduces challenges in coherence, planning, and discourse modeling (Kumar et al., 2024; Que et al., 2024; Wu et al., 2025). Techniques include retrieval-augmented generation (Lewis et al., 2021), self-refinement (Madaan et al., 2023), and iterative lengthening (Quan et al., 2024). Novel generation demands plot and character consistency over long spans (Xie et al., 2023; Yao et al., 2019; Guan et al., 2021). Hierarchical planning (Wang et al., 2023; Yang et al., 2023; Wang et al., 2024), memory-based agents (Bai et al., 2024), and reinforcement-based control (Doshi and Hauser, 2024; Chhun) have been explored, but ultra-long contexts remain difficult.

LLM-based compression. Some work treats LLMs as adaptive entropy coders (Shin), e.g., LLMZIP (Valmeekam et al., 2023) and DEEPZIP (Goyal et al., 2018), achieving near-Shannon compression via token probability coding. We instead explore lossy but semantically faithful compression: can an LLM expand a sparse outline into a coherent million-word novel? We analyze the acceptable range of outline entropy H_{outline} that balances compression with reconstructive fidelity.

LLM as judge. We assess both (i) distortion from the original novel and (ii) the quality of generated output. Distortion metrics include traditional algorithms (Cosine, Hamming, Jaccard, Levenshtein, TF-IDF) (Gahman and Elangovan, 2023a; Gomaa and Fahmy, 2013; Singh and Singh, 2020; Wang and Dong, 2020; Gahman and Elangovan, 2023b) and pre-trained models (BERTScore, BLEU, ROUGE, METEOR) (Guan et al., 2022; Xiao et al., 2024; Yu et al., 2024; Yuan et al., 2023). Novel quality is evaluated via aesthetic score, coherence, creativity, consistency, and fluency (Ichmoukhamedov et al., 2024; Venkatraman et al., 2024), with recent studies proposing LLMs themselves as evaluators (Gu et al., 2025).

2 Methodology

2.1 Notation and Two-Stage Pipeline.

Let $X_0 \in \mathcal{V}^{L_0}$ denote the original novel with L_0 words. A sequence of K compression mappings

$$\{\phi_i : \mathcal{V}^{L_{i-1}} \to \mathcal{V}^{L_i}\}_{i=1}^{K}$$

produces progressively shorter outlines

$$X_i = \phi_i(X_{i-1}), \qquad \alpha_i = \frac{L_i}{L_0}, \qquad i = 1, \ldots, K,$$

where $X_i \in \mathcal{V}^{L_i}$ and α_i is the cumulative compression ratio after level i.

Restoration applies the corresponding expansion mappings

$$\{\psi_i : \mathcal{V}^{L_i} \to \mathcal{V}^{L_{i-1}}\}_{i=K}^{1} : \quad \hat{X}_{i-1} = \psi_i(\hat{X}_i),$$

with initials

$$\hat{X}_K = X_K, \qquad \hat{X} = \hat{X}_0.$$

The overall objective, following rate–distortion theory(Blau and Michaeli, 2019), is

$$\min_{\{\phi_i, \psi_i\}_{i=1}^{K}} R = \frac{H(X_K)}{L_0} \quad \text{s.t.} \quad D(\hat{X}, X_0) \leq \epsilon,$$

where $H(\cdot)$ is defined as the number of words of text(Bentz et al., 2017) and $D(\cdot, \cdot)$ is a task-specific distortion bounded by ϵ in our study.

Outline–Expansion Pipeline. We implement three pipelines, as shown in Figure 1: **(i)Single-Stage:** generated from the full novel, then expanded directly to explore distortion under extreme compression. **(ii)Hierarchical Two-Stage:** The compression first reduces the original novel X_0 to a high-level outline X_1 with rate α_1, and then further compresses it to a global summary X_2 with rate α_2. The expansion reconstructs $\hat{X}_1$ from X_2, and finally recovers $\hat{X}_0$ from $\hat{X}_1$.

(iii)Mixed Two-Stage: The compression is the same as above. However, the expansion skips the intermediate outline and directly reconstructs $\hat{X}_0$ from X_2, under varying compression ratios R.

We can derive the compression ratio to be $R = \alpha_1 \times \alpha_2$.

Using traditional methods like LZ77, lossless compression achieves $R \approx 0.5$ (Amir et al., 2024), which serves as a lower bound.

We define distortion as a composite of Cosine distance, BERTScore, and Euclidean distance of detail counts, detailed in our evaluation metrics.

2.2 Dataset Configuration

We sample 40 public-domain Chinese novels (10 each from Fantasy, Urban, Romance, and Historical genres) from dataset (zxbsmk, 2023). Each 1M-word novel is split into 5,000-word chapters for reliable processing.

Compression/expansion prompts are detailed in Appendix C–D. Translation experiments confirm language versions have minimal impact.

For evaluation (Appendix J), we generate 200 section and one global outline per novel. Eight chapters per novel are reconstructed for comparison.

2.3 Controlled Variables.

Our configuration space is defined by five independent factors: (i) the *novel genre* selected for evaluation; (ii) the *compression ratio* applied at each outline level (global and section); (iii) the *LLM model*, which is the backbone model.

2.4 Evaluation Metrics

To quantify reconstruction distortion, we define a three distance function $D(X, \hat{X})$ as:

$D_{\text{trad}}(X, \hat{X})$ combines one minus the cosine similarity score and one minus BERTScore between embeddings of original and reconstructed novels;

$D_{\text{llm}}(X, \hat{X})$ is the average GPT-4o judgment score across semantic, plot, character, background, and style similarity;

$D_{\text{struct}}(X, \hat{X})$ measures absolute differences in counts of unique characters, scenes, and items.

3 Experiment

We evaluate hierarchical compression–expansion strategies for ultra-long novel generation using **Gemini 2.0 Flash** (temperature = 0.3), focusing on: (1) How do different compression ratios affect semantic fidelity? (2) Does a two-stage outline hierarchy ($K = 2$) outperform a single-stage ($K = 1$)?

Gemini is chosen for its large context window and throughput efficiency. Each novel involves near 3M input and 500K output tokens.

3.1 Baseline Setup ($K = 1$)

We test a single-stage pipeline using a **10,000-word outline** ($R = \alpha_1 \approx 0.01$) generated from the full novel, then expanded directly to explore distortion under extreme compression.

We compare four methods: **(A)** Human-written original, **(B)** Translate → Reconstruct, **(C)** Direct

Setting	$\mathbf{R = \alpha_1 \times \alpha_2}$	Cosine	BERT F1	SemSim	CharSim	StyleSim	CharDiff	SceneDiff	PropDiff
B	1.000	0.922 ± 0.054	0.602 ± 0.085	0.912 ± 0.081	0.914 ± 0.104	0.919 ± 0.096	0.50 ± 1.18	0.36 ± 0.99	0.44 ± 1.00
C	0.010	0.645 ± 0.069	0.169 ± 0.038	0.418 ± 0.179	0.410 ± 0.195	0.350 ± 0.160	7.36 ± 5.77	3.85 ± 3.75	4.65 ± 4.89
D	0.010	0.633 ± 0.077	0.160 ± 0.045	0.390 ± 0.209	0.420 ± 0.202	0.305 ± 0.191	7.77 ± 4.96	5.11 ± 3.38	6.24 ± 5.05
K2-*	0.010	$\mathbf{0.677 \pm 0.073}$	$\mathbf{0.199 \pm 0.046}$	$\mathbf{0.613 \pm 0.216}$	$\mathbf{0.566 \pm 0.201}$	$\mathbf{0.611 \pm 0.133}$	$\mathbf{6.70 \pm 5.89}$	$\mathbf{4.13 \pm 4.07}$	$\mathbf{4.66 \pm 5.27}$
K2-0	0.001	0.637 ± 0.068	0.169 ± 0.040	0.439 ± 0.201	0.289 ± 0.164	0.493 ± 0.139	10.19 ± 6.53	5.10 ± 3.79	4.88 ± 5.08
K2-1	0.005	0.661 ± 0.077	0.189 ± 0.047	0.591 ± 0.234	0.474 ± 0.189	0.600 ± 0.147	8.62 ± 5.75	4.65 ± 4.06	4.88 ± 4.63
K2-2	0.010	0.665 ± 0.074	0.188 ± 0.047	0.578 ± 0.232	0.488 ± 0.202	0.581 ± 0.145	8.70 ± 6.04	4.63 ± 3.74	5.16 ± 5.87
K2-3	0.001	0.650 ± 0.070	0.168 ± 0.041	0.517 ± 0.210	0.354 ± 0.183	0.538 ± 0.137	9.88 ± 6.26	4.97 ± 3.84	5.02 ± 4.88
K2-4	0.005	0.649 ± 0.074	0.179 ± 0.038	0.572 ± 0.217	0.452 ± 0.169	0.575 ± 0.134	9.30 ± 6.26	4.81 ± 3.97	5.20 ± 5.61
K2-5	0.010	0.655 ± 0.074	0.181 ± 0.043	0.549 ± 0.231	0.458 ± 0.192	0.572 ± 0.147	9.15 ± 6.38	4.87 ± 3.92	$\mathbf{4.66 \pm 5.72}$
K2-6	0.015	0.668 ± 0.073	0.193 ± 0.041	0.667 ± 0.184	0.535 ± 0.167	0.628 ± 0.119	8.76 ± 6.56	4.98 ± 3.94	5.03 ± 6.07
K2-7	0.020	0.668 ± 0.07	0.192 ± 0.044	0.71 ± 0.166	0.583 ± 0.158	0.654 ± 0.110	9.53 ± 7.58	5.09 ± 4.30	5.42 ± 7.00
K2-8	0.015	0.674 ± 0.075	0.198 ± 0.043	0.67 ± 0.181	0.531 ± 0.168	0.634 ± 0.116	8.47 ± 6.59	4.63 ± 3.81	4.49 ± 4.11
K2-9	0.020	0.679 ± 0.073	0.197 ± 0.045	0.663 ± 0.193	0.543 ± 0.202	0.633 ± 0.125	9.41 ± 6.23	5.17 ± 4.15	5.31 ± 5.03

Table 2: Grouped statistics (mean $\pm$ std) for similarity and structural difference metrics, excluding the baseline (B), to identify best-performing configurations. The compression ratio is computed as $R = \alpha_1 \times \alpha_2$. Bolded values indicate the best average performance for each metric among the tested configurations under $R \leq 0.01$, but do not imply statistical significance. Formal significance testing results are presented in a separate table. Distortion is computed as one minus similarity.

compression–expansion, and **(D)** LongWriter.

3.2 Hierarchical Setup ($K = 2$)

In the two-stage pipeline, the novel X_0 is compressed into a section-level outline X_1, then further into a global outline X_2, enabling multi-resolution control.

Compression ratios $\alpha_1 \in \{0.05, 0.10\}$, $\alpha_2 \in \{0.01, 0.05, 0.10, 0.20, 0.30, 0.40\}$ determine abstraction levels (Table 3).

ID	α_1	α_2	L_1	L_2
K2-0	0.05	0.01	50,000	1,000
K2-1	0.05	0.10	50,000	5,000
K2-3	0.05	0.20	50,000	10,000
K2-4	0.10	0.01	100,000	1,000
K2-5	0.10	0.05	100,000	50,000
K2-6	0.05	0.30	50,000	15,000
K2-7	0.05	0.40	50,000	20,000
K2-8	0.10	0.15	100,000	15,000
K2-9	0.10	0.20	100,000	20,000
K2-*	0.05	0.20	50,000	10,000

Table 3: Two-stage compression configurations under Gemini-2.0-Flash. The method K2-* is first apply two stage compressions, then apply direct expansion.

Outlines is structured using JSON templates paragraph summaries.

4 Analysis

4.1 Significance Testing

Pairwise t-tests show that most compression configurations differ significantly in distortion ($p < 0.05$), especially in character similarity. Notably, **K2-*** outperforms all settings with $R \leq 0.01$ ($p < 0.001$), while **K2-3** underperforms compared to **K2-4** and **K2-5**. Some comparisons (e.g., K2-1 vs. K2-2, $p = 0.60$) show no significant difference, suggesting robustness in certain ranges. **B** confirms that translated text remains semantically close to the original.

Distortion decreases with higher compression ratio, but gains from $R = 0.001$ to 0.01 are larger than those from 0.01 to 0.02.

Visualization. Appendix figures show: (i) correlation between R and similarity (Figure 3); (ii) grouped statistics (Table 3); (iii) significance heatmap (Figure 2).

Sampling Justification. Although each novel has about 200 chapters, we sample 8 chapters per novel. A pilot study over 40 books shows a Pearson correlation $r = 0.95$ between full-book distortion and sampled distortion.

Fisher z-transform analysis confirms that $r > 0.90$ even under 95% confidence, validating the proxy method.

5 Conclusion

Using **Gemini 2.0 Flash**, we find that structured JSON outlines with $\alpha_1 = 0.05$, $\alpha_2 = 0.20$, and direct expansion (**K2-***) yield the best fidelity to original novels in both semantic and structural metrics.

6 Discussion

We hypothesize that section-level outlines improve compression by guiding localized abstraction, but may be less necessary in expansion due to the global outline's context coverage. The weak correlation between R and similarity likely reflects the model's ability to exploit global context, reducing reliance on fine-grained section summaries.

Limitations

This study has several limitations. First, the proposed hierarchical framework may face challenges when applied to novels with intricate or nonlinear structures, such as mysteries. Second, all experiments were conducted on Chinese texts, which may limit the generalizability of our findings across cultural contexts—a factor that extends beyond language alone. Third, the framework incurs substantial computational costs, potentially constraining its scalability. Selecting $H(\cdot)$ to be the number of words of text is a practical way compared to computing Shannon entropy of text. Finally, although care was taken to ensure fair evaluation, the use of LLM-based scoring may introduce systematic biases. In practical applications, human evaluation may serve as a more accurate and context-aware assessment method.

During compression and reconstruction, we acknowledging that all two stages incur token costs. However, we exclude $\sum_{k=1}^{2} H(X_k) = \alpha_1 \times \alpha_2 + \alpha_1$ from consideration, as our focus is on generating ultra-long novels from global outlines. We do not fix a specific value of ϵ, as our study demonstrates that differences in performance across compression ratios are significant. However, there is insufficient evidence to justify any particular choice of ϵ as universally optimal.

Ethical Considerations

We exclusively chose publicly available sources (zxbsmk, 2023) and Chinese version of ultra-long novels from Project Gutenberg for evaluation. We follow the claim 'This dataset and any derivatives generated from it may be used for research purposes only. Commercial use and any other applications that may cause harm to society are strictly prohibited.' Any data contains offensive content has been filtered by Gemini 2.0 flash. Given our result will be an optimal hyperparameter and no pretrained model or dataset will be provided to the public, the risk of ethical concerns is minimal. However, we should also consider that the use of language models in long-form creative writing may impact authors' livelihoods and raise concerns about bias and the propagation of misinformation.

Acknowledgments

We gratefully acknowledge the use of AI-assisted tools solely for grammatical corrections during manuscript preparation. No other aspects of the research— including conceptualization, experimental design, data analysis, or interpretation of results—were generated or modified by AI. All substantive content and conclusions were developed independently by the authors.

Implementation Details

The code has been published at anonymous space.

References

Amihood Amir, Itai Boneh, Panagiotis Charalampopoulos, and Sarel Klein. 2024. Streaming Pattern Matching with Wildcards. In *35th Annual Symposium on Combinatorial Pattern Matching (CPM 2024)*, volume 296 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 3:1–3:14, Dagstuhl, Germany. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

J. Bai, S. Bai, Y. Chu, Z. Cui, K. Dang, X. Deng, Y. Fan, W. Ge, Y. Han, F. Huang, B. Hui, L. Ji, M. Li, J. Lin, R. Lin, D. Liu, G. Liu, C. Lu, K. Lu, and 1 others. 2023. Qwen technical report. *Preprint*, arXiv:2309.16609.

Y. Bai, J. Zhang, X. Lv, L. Zheng, S. Zhu, L. Hou, Y. Dong, J. Tang, and J. Li. 2024. Longwriter: Unleashing 10,000+ word generation from long context llms. *Preprint*, arXiv:2408.07055.

Christian Bentz, Dimitrios Alikaniotis, Michael Cysouw, and Ramon Ferrer-i Cancho. 2017. The entropy of words—learnability and expressivity across more than 1000 languages. *Entropy*, 19(6):275.

Yochai Blau and Tomer Michaeli. 2019. Rethinking lossy compression: The rate-distortion-perception tradeoff. *Preprint*, arXiv:1901.07821.

C. Chhun. Meta-evaluation methodology and benchmark for automatic story generation. Accessed: 2025-04-28.

DeepSeek-AI, A. Liu, B. Feng, B. Xue, B. Wang, B. Wu, C. Lu, C. Zhao, C. Deng, C. Zhang, C. Ruan, D. Dai, D. Guo, D. Yang, D. Chen, D. Ji, E. Li, F. Lin, F. Dai, and 1 others. 2025. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

Anil R. Doshi and Oliver P. Hauser. 2024. Generative AI enhances individual creativity but reduces the collective diversity of novel content. *Science Advances*, 10(28).

N. Gahman and V. Elangovan. 2023a. A comparison of document similarity algorithms. *International Journal of Artificial Intelligence & Applications*, 14(2):41–50.

N. Gahman and V. Elangovan. 2023b. A comparison of document similarity algorithms. *Preprint*, arXiv:2304.01330.

Wael H. Gomaa and Aly A. Fahmy. 2013. A survey of text similarity approaches. *International Journal of Computer Applications*, 68(13).

M. Goyal, K. Tatwawadi, S. Chandak, and I. Ochoa. 2018. Deepzip: Lossless data compression using recurrent neural networks. *Preprint*, arXiv:1811.08162.

A. Grattafiori, A. Dubey, A. Jauhri, A. Pandey, A. Kadian, A. Al-Dahle, A. Letman, A. Mathur, A. Schelten, A. Vaughan, A. Yang, A. Fan, A. Goyal, A. Hartshorn, A. Yang, A. Mitra, A. Sravankumar, A. Korenev, A. Hinsvark, and 1 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

J. Gu, X. Jiang, Z. Shi, H. Tan, X. Zhai, C. Xu, W. Li, Y. Shen, S. Ma, H. Liu, S. Wang, K. Zhang, Y. Wang, W. Gao, L. Ni, and J. Guo. 2025. A survey on llm-as-a-judge. *Preprint*, arXiv:2411.15594.

J. Guan, X. Mao, C. Fan, Z. Liu, W. Ding, and M. Huang. 2021. Long text generation by modeling sentence-level and discourse-level coherence. *Preprint*, arXiv:2105.08963.

Jie Guan, Zichao Feng, Yihong Chen, Ruqing He, Xiaoyan Mao, Chang Fan, and Minlie Huang. 2022. Lot: A story-centric benchmark for evaluating chinese long text understanding and generation. *Preprint*, arXiv:2108.12960.

T. Ichmoukhamedov, J. Hinns, and D. Martens. 2024. How good is my story? towards quantitative metrics for evaluating llm-generated xai narratives. *Preprint*, arXiv:2412.10220.

I. Kumar, S. Viswanathan, S. Yerra, A. Salemi, R. A. Rossi, F. Dernoncourt, H. Deilamsalehy, X. Chen, R. Zhang, S. Agarwal, N. Lipka, C. V. Nguyen, T. H. Nguyen, and H. Zamani. 2024. Longlamp: A benchmark for personalized long-form text generation. *Preprint*, arXiv:2407.11016.

P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, T. Rocktäschel, S. Riedel, and D. Kiela. 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Preprint*, arXiv:2005.11401.

Junyi Li, Tianyi Tang, Wayne Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2022. Pretrained language models for text generation: A survey.

A. Madaan, N. Tandon, P. Gupta, S. Hallinan, L. Gao, S. Wiegreffe, U. Alon, N. Dziri, S. Prabhumoye, Y. Yang, S. Gupta, B. P. Majumder, K. Hermann, S. Welleck, A. Yazdanbakhsh, and P. Clark. 2023. Self-refine: Iterative refinement with self-feedback. *Preprint*, arXiv:2303.17651.

N. Mikhaylovskiy. 2023. Long text generation challenge. *Preprint*, arXiv:2306.02334.

Olaniyi Mathew Olayiwola, Fadeke Sola Apantaku, Hammed Oladiran Bisira, and Adedayo Amos Adewara. 2013. Efficiency of neyman allocation procedure over other allocation procedures in stratified random sampling. *American Journal of Theoretical and Applied Statistics*, 2(5):122–127.

OpenAI, A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. J. Ostrow, A. Welihinda, A. Hayes, A. Radford, A. Mądry, A. Baker-Whitcomb, A. Beutel, A. Borzunov, A. Carney, A. Chow, A. Kirillov, A. Nichol, and 1 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

S. Quan, T. Tang, B. Yu, A. Yang, D. Liu, B. Gao, J. Tu, Y. Zhang, J. Zhou, and J. Lin. 2024. Language models can self-lengthen to generate long texts. *Preprint*, arXiv:2410.23933.

H. Que, F. Duan, L. He, Y. Mou, W. Zhou, J. Liu, W. Rong, Z. M. Wang, J. Yang, G. Zhang, J. Peng, Z. Zhang, S. Zhang, and K. Chen. 2024. Hellobench: Evaluating long text generation capabilities of large language models. *Preprint*, arXiv:2409.16191.

D. Shin. Better text compression using a large language model. *Defensive Publications Series*.

Ritika Singh and Satwinder Singh. 2020. Text similarity measures in news articles by vector space model using nlp. *Journal of the Institution of Engineers (India): Series B*.

Yixuan Su, Tian Lan, Yan Wang, Dani Yogatama, Lingpeng Kong, and Nigel Collier. 2022. A contrastive framework for neural text generation. *Preprint*, arXiv:2202.06417.

G. Team, R. Anil, S. Borgeaud, J.-B. Alayrac, J. Yu, R. Soricut, J. Schalkwyk, A. M. Dai, A. Hauth, K. Millican, D. Silver, M. Johnson, I. Antonoglou, J. Schrittwieser, A. Glaese, J. Chen, E. Pitler, T. Lillicrap, A. Lazaridou, and 1 others. 2024. Gemini: A family of highly capable multimodal models. *Preprint*, arXiv:2312.11805.

Chinthalapeta Sri Krishna Valmeekam, Karthikeyan Narayanan, Dileep Kalathil, Jean-François Chamberland, and Sanjay Shakkottai. 2023. Llmzip: Lossless text compression using large language models. *Preprint*, arXiv:2306.04050.

S. Venkatraman, A. Uchendu, and D. Lee. 2024. Gpt-who: An information density-based machine-generated text detector. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 103–115.

Jing Wang and Yu Dong. 2020. Measurement of text similarity: A survey. *Information*, 11(9):Article 9.

Q. Wang, J. Hu, Z. Li, Y. Wang, D. Li, Y. Hu, and M. Tan. 2024. Generating long-form story using dynamic hierarchical outlining with memory-enhancement. *Preprint*, arXiv:2412.13575.

Y. Wang, K. Yang, X. Liu, and D. Klein. 2023. Improving pacing in long-form story planning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10788–10845.

S. Wu, Y. Li, X. Qu, R. Ravikumar, Y. Li, T. Loakman, S. Quan, X. Wei, R. Batista-Navarro, and C. Lin. 2025. Longeval: A comprehensive analysis of long-text generation through a plan-based paradigm. *Preprint*, arXiv:2502.19103.

Shuo Xiao, Ziyang Liu, Peng Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 641–649.

Z. Xie, T. Cohn, and J. H. Lau. 2023. The next chapter: A study of large language models in storytelling. In *Proceedings of the 16th International Natural Language Generation Conference*, pages 323–351.

K. Yang, D. Klein, N. Peng, and Y. Tian. 2023. Doc: Improving long story coherence with detailed outline control. *Preprint*, arXiv:2212.10077.

L. Yao, N. Peng, R. Weischedel, K. Knight, D. Zhao, and R. Yan. 2019. Plan-and-write: Towards better automatic storytelling. *Preprint*, arXiv:1811.05701.

Ling Yu, Bo Liu, Qian Lin, Xian Zhao, and Cheng Che. 2024. Semantic similarity matching for patent documents using ensemble bert-related model and novel text processing method. *Preprint*, arXiv:2401.06782.

Li Yuan, Shuai Gao, and Peng Pan. 2023. Ctsarf: A chinese text similarity analysis model based on residual fusion. *Neurocomputing*, 559:126801.

zxbsmk. 2023. Webnovel cn dataset. https://huggingface.co/datasets/zxbsmk/webnovel_cn. Accessed: 2025-06-28.

A Appendix: Pairwise significance test results between compression configurations

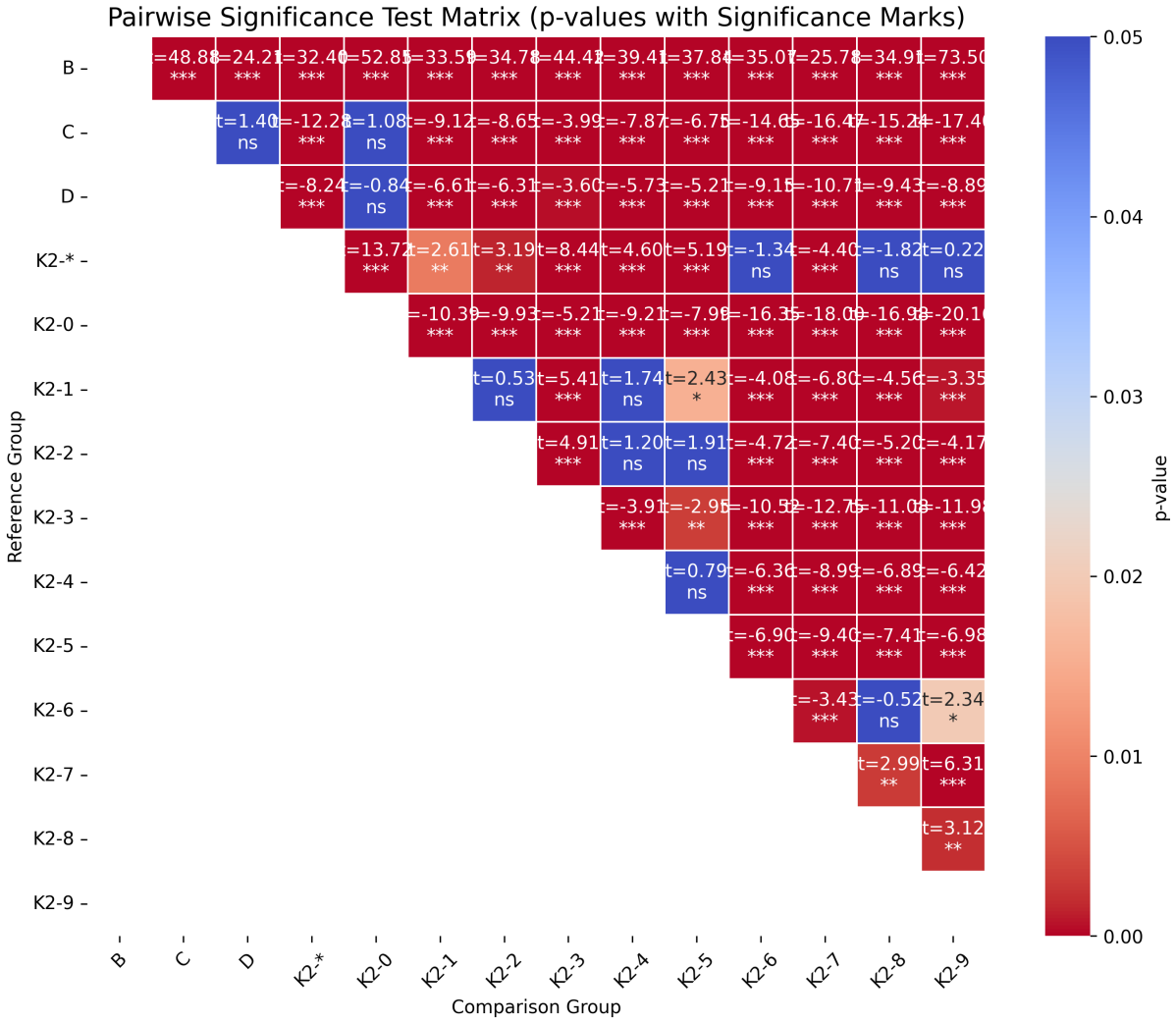

Figure 2: Pairwise significance test results between compression configurations. Each cell shows the t-value and significance level (* for $p < 0.05$, ** for $p < 0.01$, *** for $p < 0.001$).

B Appendix: Correlation Between R and Mean Similarity Excluding Group BD

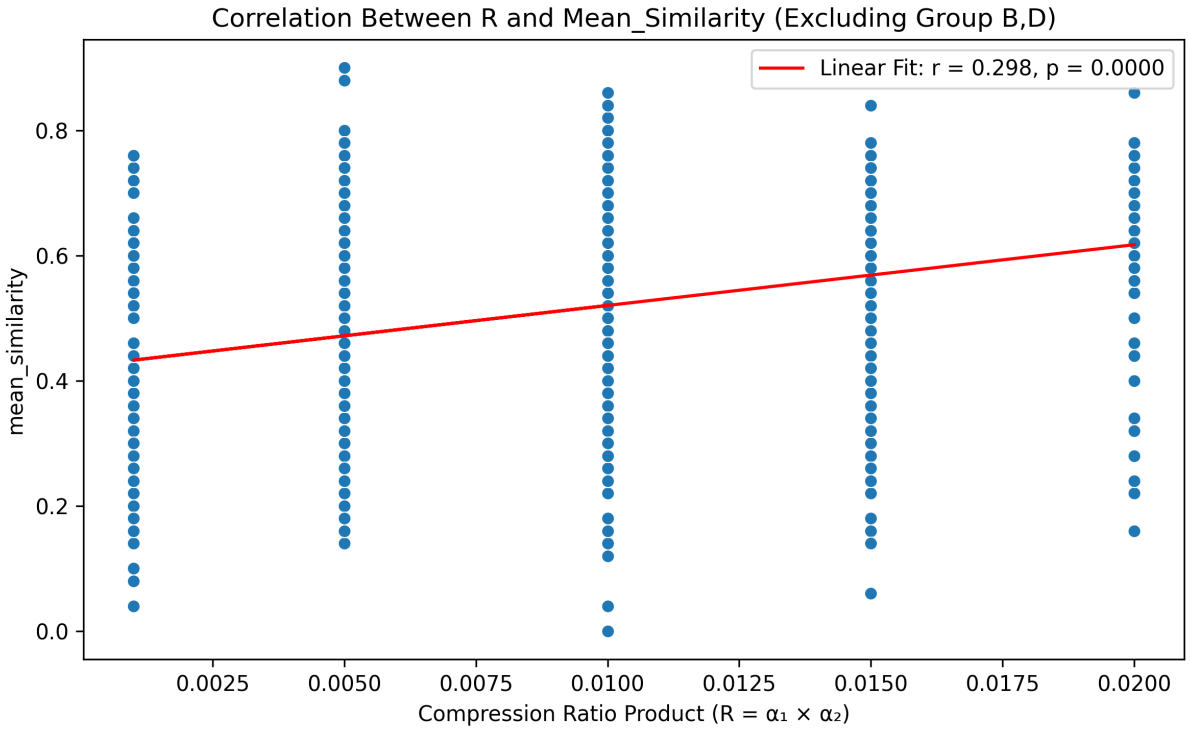

Figure 3: Correlation between the compression ratio product ($R = \alpha_1 \times \alpha_2$) and the average similarity score, computed as the mean of semantic similarity, plot similarity, character similarity, background similarity, and style similarity, excluding the baseline group B,D. A weak but statistically significant positive correlation is observed ($r = 0.129$, $p < 0.001$).

C Appendix: Prompts Templates: LongWriter baseline

Chinese Prompt Template (LongWriter Reconstruction):
根据以下【小说大纲】写第{chap_num}章，要求：
1. 字数在5000字。
2. 文笔流畅，语言生动。

【小说大纲】：{outline_text}

【第{chap_num}章】正文开始：

English Translation:
Based on the following [NOVEL OUTLINE], write Chapter {chap_num}.
Requirements:
1. Word count: approximately 5,000 words.
2. Maintain fluent writing style and vivid language.

【NOVEL OUTLINE】: {outline_text}

【Chapter {chap_num}】 begins:

Figure 4: Prompt templates used for LongWriter baseline reconstruction. English Translation is used for novels written in English. Chinese Translation is used for novels written in Chinese.

D Appendix: Prompts Templates: Hierarchical Prompts: K=2 Stage 1 Compression

Chinese Prompt Template (Chapter Analysis):

你是一位专业的文学编辑。请仔细阅读我提供的【章节全文】。
你的任务是提取结构化信息，并 **必须** 调用 ‘extract_chapter_details‘ 函数来返回结果。
请严格按照函数参数的描述（特别是关于"情节摘要导语"的详细程度要求）来填充信息。
绝对不要 输出任何 JSON 格式之外的文本、解释、代码块标记（如 “‘json ... “‘）或 Markdown。
直接调用函数并填充其参数。

【章节全文】：{chapter_text}

English Translation:

You are a professional literary editor. Please carefully read the [CHAPTER TEXT] I provide.
Your task is to extract structured information and **must** call the ‘extract_chapter_details‘ function to return results.
Please strictly follow the function parameter descriptions (especially the detailed requirements for "plot summary introduction") to fill in the information.
Never output any text, explanations, code block markers (like “‘json ... “‘) or Markdown outside of JSON format.
Directly call the function and fill in its parameters.

【CHAPTER TEXT】: {chapter_text}

JSON Schema Structure:

```
{
  "情节摘要导语": "string", // Plot summary introduction
  "出现人物": ["string"], // Characters appearing
  "出现道具": ["string"], // Props appearing
  "出现场景": ["string"], // Scenes appearing
  "伏笔_设下": ["string"], // Foreshadowing set
  "伏笔_回收": ["string"] // Foreshadowing resolved
}
```

Figure 5: Prompt templates used for our hierarchical method’s K=2 stage 1 compression to section outline. The JSON schema ensures consistent extraction of narrative elements across all experimental conditions.

E Appendix: Prompts Templates: Hierarchical Prompts: K=2 Stage 2 Compression

Chinese Prompt Template (Compression K=2 Stage2):

你是一位资深中文小说编辑，请将下面的【整书细纲】精炼为总字数约10000字的大纲。
【输出要求】
- 只用简体中文，以第几章节：大纲情节的格式输出，不要章节这两个字，保留章节号；
- 保持情节完整，突出主要人物、冲突、转折、结局；
- 不得输出任何额外解释或注释。
- 每个段落要精简，必须涵盖从第一章到最后一章的全部细纲内容。
约10000字，及每章约40字。200章总结完即停止输出。

【整书细纲】：{detailed_outline}

English Translation:

You are a senior Chinese novel editor. Please refine the following [DETAILED BOOK OUTLINE] into an outline of approximately 10,000 words.
【Output Requirements】
- Use only Simplified Chinese, output in the format "Chapter X: plot outline", omit the word "章节", keep chapter numbers;
- Maintain plot integrity, highlight main characters, conflicts, turning points, and endings;
- Do not output any additional explanations or annotations.
- Each paragraph should be concise and must cover all detailed outline content from the first chapter to the last chapter.
Approximately 10,000 words, about 40 words per chapter. Stop output after summarizing 200 chapters.

【DETAILED BOOK OUTLINE】: {detailed_outline}

Figure 6: Prompt templates used for hierarchical compression stage (K=2, Stage2). This template compresses detailed chapter outlines into a concise 10,000-word summary while preserving narrative structure. 10,000 words is a variable.

F Appendix: Prompts Templates: Hierarchical Prompts: K=2 Stage 2 Expansion

Chinese Prompt Template (Expansion K=2 Stage2):

请根据大纲的对应章节写一段 200-300 字的情节摘要导语，字数必须在这个区间，不得超出也不得少于。结尾不要加字数统计
你是一位专业中文小说策划。下面给出【整书章节大纲】。
请扩写并输出 **第 {n} 章** 的结构化细纲。
必须调用函数 ‘extract_chapter_details‘ 按参数要求返回结果，
绝不能输出 JSON 之外的任何文字或 Markdown。

【整书章节大纲】：{outline}

English Translation:

Please write a 200-300 word plot summary introduction for the corresponding chapter in the outline. The word count must be within this range, neither exceeding nor falling short. Do not add word count statistics at the end.
You are a professional Chinese novel planner. The [COMPLETE BOOK CHAPTER OUTLINE] is given below.
Please expand and output the structured detailed outline for **Chapter {n}**.
Must call the function ‘extract_chapter_details‘ to return results according to parameter requirements, Never output any text or Markdown outside of JSON.

【COMPLETE BOOK CHAPTER OUTLINE】: {outline}

JSON Schema Structure:

```
{
  "情节摘要导语": "string", // Plot summary introduction (400-500 words)
  "出现人物": ["string"], // Characters appearing
  "出现道具": ["string"], // Props appearing
  "出现场景": ["string"], // Scenes appearing
  "伏笔_设下": ["string"], // Foreshadowing set
  "伏笔_回收": ["string"] // Foreshadowing resolved
}
```

Figure 7: Prompt templates used for hierarchical expansion stage (K=2, Stage2). This template expands compressed outlines into structured 50,000-word detailed chapter outlines with specific narrative elements. The 200-300 word requirement ensures around 50,000 words in total.

G Appendix: Prompts Templates: K=1 Direct Compression Method

Chinese Prompt Template (Direct Compression):
你是一位资深中文小说编辑，现在需要为整本书撰写一份【整书大纲】。

【输出规范】
1. 全文仅使用简体中文;
2. 字数 ⩽1000 汉字;
3. 完整概括出主线剧情故事，尤其是主要人物、核心冲突、关键转折与结局;
4. 不要出现章节标题、序号、列表符号，直接以自然段叙述;
5. 开头不得使用"以下是"或类似提示语，应直接进入正文。
6. 概括全文，注意，是概括100万字小说的从开头到结尾的故事。
7. 概括全文的同时保留尽量多的细节，尽量多的人物，尽量多的重要情节
8. 告诉我最后一章节的标题，这个部分不算在1000字的限制内，作为你阅读了整本书的测试

请严格遵守以上规则，一次性输出完成后的整书概要。

【完整小说文本】：{full_novel_text}

English Translation:
You are a senior Chinese novel editor, and now you need to write a [COMPLETE BOOK OUTLINE] for the entire book.

【Output Specifications】
1. Use only Simplified Chinese throughout;
2. Word count ⩽1000 Chinese characters;
3. Completely summarize the main storyline, especially main characters, core conflicts, key turning points and endings;
4. Do not include chapter titles, numbers, or list symbols, narrate directly in natural paragraphs;
5. Do not start with "The following is" or similar prompts, should directly enter the main text.
6. Summarize the full text, note that this is to summarize a 1 million word novel from beginning to end.
7. While summarizing the full text, retain as many details, characters, and important plots as possible
8. Tell me the title of the last chapter, this part does not count towards the 1000-word limit, as a test of your reading of the entire book

Please strictly follow the above rules and output the completed book summary in one go.

【COMPLETE NOVEL TEXT】: {full_novel_text}

Figure 8: Prompt templates used for direct compression method. This approach directly compresses the complete novel (1 million words) into a concise 1000-word outline while preserving essential narrative elements, characters, and plot details. The last chapter title requirement serves as a verification mechanism.

H Appendix: Prompts Templates: K=2 Mixed Hierarchical Direct Expansion and K=1 Expansion

Chinese Prompt Template (Mixed Hierarchical Direct Expansion):
你是一位擅长情节创作的中文作家，现在需要根据【整书大纲】扩写第 {chap_num} 章。

【整书大纲】
{outline_text}

【写作要求】
1. 语言生动连贯；
2. 字数绝对不要少于 5000 字；
3. 聚焦本章情节；
4. 只输出正文，无标题。

English Translation:
You are a Chinese writer skilled in plot creation. Now you need to expand Chapter {chap_num} based on the [COMPLETE BOOK OUTLINE].

【COMPLETE BOOK OUTLINE】
{outline_text}

【Writing Requirements】
1. Vivid and coherent language;
2. Word count must not be less than 5000 words;
3. Focus on this chapter's plot;
4. Output only the main text, no title.

Figure 9: Prompt templates used for mixed hierarchical direct expansion method. This approach directly expands from compressed outline to full chapter content (5000+ words) without intermediate structured analysis, providing a streamlined generation process while maintaining narrative quality.

I Appendix: Prompts Templates: LLM-based Evaluation

Chinese Prompt Template (LLM Evaluation):

你是一位专业中文小说编辑，请你阅读【文本A】与【文本B】，完成以下任务：
1. 分别提取文本A与文本B中的：
- 出现道具列表（如：剑、玉、令牌等）
- 出现人物名称列表
- 出现场景/环境名称列表
* 提取时请尽量精确，去除通用词语 (例如: '人', '地方')，只保留具体名称。
* 如果某一项在文本中没有出现，请返回空列表 '[]'。
2. 分别统计每类元素的数量（去重后），并输出每类的元素列表与数量。
3. 接着对比两段文本内容，按照以下 5 个维度进行 0-1 评分（1 表示非常相似，0 表示完全不同）：
- semantic_similarity 整体语义/主题
- plot_similarity 情节、事件发展
- character_similarity 人物名称、数量与设定（综合考虑）
- background_similarity 场景与世界设定
- style_similarity 语言风格与表达方式
* 评分请基于文本内容，给出客观评估。
请严格输出以下 JSON 格式，不要包含 markdown "'json ... "' 标记，直接输出 JSON 对象：
【文本A】{text_a}
【文本B】{text_b}

English Translation:

You are a professional Chinese novel editor. Please read [TEXT A] and [TEXT B] and complete the following tasks:
1. Extract from Text A and Text B respectively:
- List of props appearing (e.g., sword, jade, token, etc.)
- List of character names appearing
- List of scene/environment names appearing
* Please extract as precisely as possible, remove generic words (e.g., 'person', 'place'), keep only specific names.
* If any category does not appear in the text, please return empty list '[]'.
2. Count the number of each type of element (after deduplication) and output the element list and count for each category.
3. Then compare the two text contents and score on the following 5 dimensions from 0-1 (1 means very similar, 0 means completely different):
- semantic_similarity Overall semantics/theme
- plot_similarity Plot and event development
- character_similarity Character names, quantity and settings (comprehensive consideration)
- background_similarity Scenes and world settings
- style_similarity Language style and expression
* Please give objective evaluation based on text content.
Please strictly output the following JSON format, do not include markdown "'json ... "' markers, output JSON object directly:
【TEXT A】 {text_a}
【TEXT B】 {text_b}

Figure 10: Prompt templates used for LLM-based evaluation across five similarity dimensions: semantic, plot, character, background, and style.

J Appendix: Sampling Design

We implement a two-stage sampling design :

We implement a two-stage sampling design:

Stage 1 (between-novel): We stratify the sampling frame into four major genres—Urban (U), Romance (R), Fantasy (F), and Historical (H)—and select n_h novels from each stratum h using probability-proportional-to-size (PPS) sampling, where the size variable is the total word count $L_i \approx 1$ million. The total sample size is set to $n = \mathbf{40}$, with allocation determined by Neyman's optimal allocation scheme (Olayiwola et al., 2013):

$$n_h = n \frac{N_h S_h}{\sum_g N_g S_g},$$

where N_h and S_h denote the number of novels and estimated standard deviation of the distortion metric within stratum h, respectively.

Stage 2 (within-novel): For each selected novel i, we treat entire chapters as secondary sampling units. We sample $m_i = \mathbf{8}$ chapters using simple random sampling without replacement (SRSWOR). If a novel contains fewer than 8 chapters, all chapters are included.

In this study, we do not incorporate sampling weights; all selected units are treated equally.