schmerle at SemEval-2026 Task 4: Exploring Large Language Model Prompting Strategies for Low-Resource Narrative Similarity Detection

Maximilian Rudolf Schmerle and Nils Constantin Hellwig
Media Informatics Group, University of Regensburg, Regensburg, Germany
maximilian.schmerle@student.ur.de
nils-constantin.hellwig@ur.de

Abstract

Narrative similarity detection has broad applications in plagiarism detection, content recommendation, and comparative narrative analysis. We present a training-free, prompting-only framework for SemEval-2026 Task 4 (Track A), which requires identifying which of two candidate stories is narratively more similar to a given anchor story. Without any fine-tuning or additional annotations, we systematically evaluate three prompt templates across five structural prompting strategies, including zero-shot and few-shot inference, narrative summarization, keyword extraction, aspect splitting, and pairwise comparison. Structured prompt templates and decomposed pairwise comparisons consistently outperform baseline configurations, achieving a peak accuracy of 72.50% on the test set and 67.75% on the final leaderboard (23th out of 44 teams).

1 Introduction

Narratives across a variety of thematic areas frequently exhibit analogous narrative patterns. The identification of these patterns is beneficial in a variety of contexts, including theoretical domains. For instance, they facilitate cross-cultural and cross-temporal narrative comparisons. In addition, there is considerable potential for practical applications in such areas as the detection of plagiarism and content recommendation systems (Pial and Skiena, 2023; Osman et al., 2012). The scientific investigation of these patterns has been an ongoing task for researchers across a variety of research fields. It ranges from defining structural invariants that facilitate comparison, such as plot structure, character roles, and thematic progression (Lévi-Strauss, 1955; Barthes and Duisit, 1975; Todorov, 1981), to formal models developed in narratology (Propp, 1968; Genette, 1980). Previous approaches to narrative understanding are diverse. A summary of tasks, such as narrative summarization and question answering, is extensively outlined in the work by Zhu et al. (2023). Solutions are also diverse. One approach is symbolic narrative understanding, which maps stories onto schemas (Schank and Abelson, 1977; Lehnert, 1981). This method enables deep reasoning and causal explanations, but lacks coverage outside known domains. To address this limitation, research shifted toward statistical event-based narrative modeling (Chambers and Jurafsky, 2008; Pichotta and Mooney, 2016). These models learn recurring event patterns from corpora such as news. Rather than predicting events, discourse and narrative structure modeling focuses on coherence in stories and employs models to predict the most coherent continuations (Barzilay and Lapata, 2005; Elson and McKeown, 2010). Large Language Models (LLMs) implicitly unify the three aforementioned paradigms, knowledge-driven AI, data-driven statistical learning, and linguistic discourse modeling, and recent advances in their reasoning and contextual abstraction capabilities have given rise to novel approaches for narrative understanding. Zhu et al. (2023) survey a broad range of narrative understanding tasks and outline how LLMs can be leveraged to address them. In an applied setting, Chun (2024) employed GPT-3.5-turbo (Brown et al., 2020) to extract narrative elements and assess semantic similarity between film stories, finding that model-generated similarity rankings closely align with human judgments.

SemEval-2026 Task 4 Track A also addresses this challenge (Hatzel et al., 2026). The shared task focuses on identifying similarities among sets of three stories. As illustrated in Figure 1, an anchor story and two candidate stories (Text A and Text B) are provided. The objective is to identify which candidate story is narratively more similar to the anchor story. To support system development, the organizers provide test, training, and sample data. For annotation, narrative similarity was assessed based on three core components: abstract theme, course of action, and story outcome.

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 2046–2056

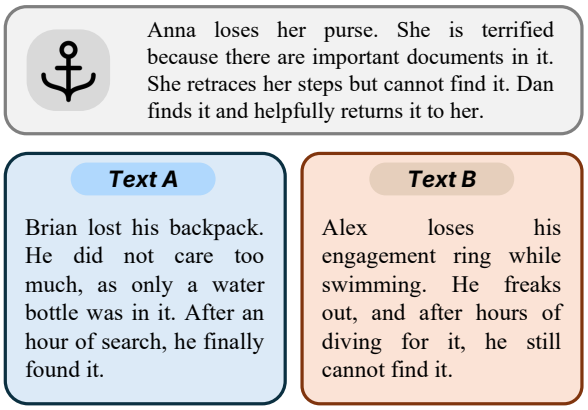

Figure 1: **SemEval-2026 Task 4 (Track A):** Given an anchor story and two candidate stories (Text A and Text B), the objective is to identify which candidate is narratively more similar to the anchor, based on abstract theme, course of action, and story outcome.

Motivated by the well-documented advantages of LLM-based approaches in terms of inference speed, cost efficiency, and scalability (Cunha et al., 2025; Sajjadi Mohammadabadi et al., 2025; Karanikolas et al., 2025), we adopt a prompting-only strategy that requires neither fine-tuning nor additional data annotation. Pretrained LLMs have shown strong performance under zero-shot and few-shot conditions (Su et al.), enabling rapid deployment across diverse domains at minimal cost (Sushil et al., 2024; Nazi et al., 2025).

We systematically evaluate three prompting strategies on Gemma 3 (27B) (Team et al., 2025) for binary narrative similarity classification. The first strategy directly adopts the task description from the shared task guidelines as a baseline. The second employs structured prompt templates with explicitly separated components, such as task context and instructions, to reduce ambiguity and improve interpretability (Wang et al., 2024; Mao et al., 2025). The third minimizes instructional complexity by distilling the prompt to its essential information, thereby reducing cognitive load on the model and sharpening task focus (Upadhayay et al., 2024; Nayab et al., 2024).

Each strategy is evaluated in both zero-shot and few-shot settings to assess the effect of in-context examples. Beyond these base configurations, we introduce two task-specific extensions targeting known prompting challenges: narrative condensation to address context length constraints, and decomposed prompting, which evaluates each of the three similarity components: abstract theme, course of action, and story outcome, in isolation. Together, these configurations constitute a comprehensive, training-free framework for narrative similarity detection.

Our experiments demonstrate that structured prompt templates and pairwise story comparisons yield the strongest performance, with the best configuration achieving 72.50% accuracy on the test set and 67.75% on the leaderboard, placing 20th out of 42 teams on the SemEval-2026 Task 4 leaderboard, competitive results obtained without any model fine-tuning or task-specific annotation.

To facilitate reproducibility and further research, we provide code and results in our GitHub repository [1].

2 System Overview

2.1 Prompting Templates

Prompt structure and semantics can notably affect model performance (Mao et al., 2025). We evaluate three templates:

Baseline Closely mirrors the wording, structure, and length of the official task description provided by the organizers, offering a comprehensive task representation as a reference point.

Structured Template Decomposes the task into explicitly labeled components, context, task, and constraints, following established prompt design principles (He et al., 2024; White et al., 2023) to improve output reliability and consistency.

Cognitive Load Reduction Strips boilerplate and verbose prose in favor of concise definitions and bullet points, improving the signal-to-noise ratio and reducing computational overhead (Zhang et al., 2025).

2.2 Prompting Approach

In addition to the prompting templates, we evaluated a range of structural prompting approaches.

Zero-Shot / Few-Shot The zero-shot condition applies the template directly to the story triple. For few-shot inference, balanced, randomly sampled human-annotated examples from the sample dataset are included in the prompt, with equal representation of Text A-closer and Text B-closer instances.

[1] https://github.com/Schmoerls/schmerle-semEval-task4-narrative-similarity

Dataset	N	Anchor			Text A			Text B		
		Avg	*Min*	*Max*	*Avg*	*Min*	*Max*	*Avg*	*Min*	*Max*
Sample Data	39	138.23	63	278	138.13	49	293	133.64	49	234
Summarization Sample Data	39	108.08	73	142	105.21	75	141	105.92	80	142
Filtering Sample Data	39	42.13	23	69	38.64	27	58	40.67	24	64
Test Data	400	137.51	46	290	138.81	39	339	138.71	39	587
Summarization Test Data	400	113.53	67	180	112.34	70	159	113.67	76	174
Filtering Test Data	400	41.71	25	73	42.05	22	83	41.84	24	83

Table 1: **Token length statistics across datasets.** We report average, minimum, and maximum token counts for Anchor, Text A, and Text B fields (computed using NLTK tokenization[2]).

Summarization To prevent long narratives from diluting model focus, all stories, including few-shot examples, were condensed into aspect-based summaries prior to classification. Summaries are generated by the same LLM as used for the classification itself, using a dedicated summarization prompt. Few-shot selection follows the procedure described above.

Keyword Extraction As a more aggressive cognitive load reduction strategy, narratives are distilled to essential keywords that preserve narrative context while eliminating all superfluous text. All story triples and few-shot examples are processed accordingly.

Aspect Splitting Each story triple is evaluated across the three similarity components (abstract theme, course of action, and outcome) in separate prompts. The three resulting classifications are aggregated via majority voting.

Pairwise Comparison Rather than presenting all three stories simultaneously, this approach decomposes the task into two separate prompts: anchor vs. Text A and anchor vs. Text B. To avoid ties, the binary output is replaced by a continuous similarity score (1–10), with the higher-scoring candidate classified as the closer narrative. When the two predictions achieved the same score, the prediction was rated as Text B being closer to the anchor text.

2.3 Output Validation

To ensure reliable and processable model outputs, we employed guided JSON generation across all prompting approaches. Rather than parsing unconstrained free-form text, we constrained the model's decoding process to produce responses conforming to a predefined JSON schema.

Split (Med.)	Field	Count	True (%)	Length: Mean ± SD
Test	Anchor Text A Text B	400	52.0	706 ± 312 (668) 714 ± 385 (694) 714 ± 676 (687)
Sample	Anchor Text A Text B	39	38.5	712 ± 289 (681) 711 ± 315 (707) 690 ± 253 (656)
Dev	Anchor Text A Text B	200	50.5	739 ± 348 (697) 714 ± 385 (703) 711 ± 344 (695)
Synthetic	Anchor Text A Text B	1,897	49.6	953 ± 320 (963) $1,031 \pm 330$ (1,030) $1,033 \pm 363$ (1,033)

Table 2: **Dataset statistics.** True (%) indicates the ratio of instances where Text A is the closer narrative. Length metrics denote characters.

3 Experimental Setup

We evaluate all combinations of the three prompt templates and five prompting approaches described above, running each configuration with three independent random seeds. For the zero-shot, few-shot, summarization, and keyword extraction approaches, we use 0, 6, 12, and 18 examples, with each condition repeated across the three seeds. Since the sample data labels were assigned based on a joint comparison of all three stories across all similarity aspects, the Aspect Splitting and Pairwise Comparison approaches were evaluated in the zero-shot setting only. Few-shot examples were drawn uniformly at random from the sample split under two constraints: (i) class balance was enforced, ensuring equal representation of Text-A-closer and Text-B-closer instances, and (ii) as the number of shots increased, additional examples were appended to smaller sets from the same seed, guaranteeing strict inclusion across shot counts.

	Identical Paper Prompt Copy				Structured Template				Cognitive Load Reduction			
	# Few-Shot Examples				*# Few-Shot Examples*				*# Few-Shot Examples*			
Task	*0*	*6*	*12*	*18*	*0*	*6*	*12*	*18*	*0*	*6*	*12*	*18*
Prompting	62.00	66.58	65.92	62.33	**67.75**	65.67	64.08	62.67	60.75	64.17	63.58	63.25
Summarize	60.50	60.50	62.50	60.42	61.75	62.33	**63.42**	60.58	60.25	62.75	61.58	60.08
Filtering	55.50	57.75	57.83	52.08	**58.50**	56.67	55.58	55.92	57.00	56.75	56.33	55.92
One to One	66.50	-	-	-	**72.50**	-	-	-	68.50	-	-	-
Aspect Splitting	61.25	-	-	-	**65.75**	-	-	-	60.75	-	-	-
Aggregate	61.15	61.61	62.08	58.28	**65.15**	61.56	61.03	59.72	61.45	61.22	60.50	59.75

Table 3: Performance comparison across three prompting strategies (*Identical Paper Prompt Copy*, *Structured Template*, *Cognitive Load Reduction*) and four few-shot configurations. The last row reports aggregate results across all tasks. **Bold** values indicate the best performance per task and shot count.

3.1 Datasets

As shown in Table 2, four subsets were released for this shared task. For all non-synthetic splits, labels were obtained via human annotation. We report performance on the test set. The sample split (39 instances) was used to construct few-shot examples, and the dev split (200 instances) served for pre-submission evaluation to identify the best-performing approach. We did not employ the LLM-generated synthetic stories and labels.

3.2 Evaluation Metrics

Following the official task guidelines, we report **accuracy** as the primary evaluation metric. Accuracy is defined as the proportion of correctly classified instances over the total number of evaluated instances:

$$\text{Accuracy} = \frac{1}{N}\sum_{i=1}^{N} \mathbb{1}(\hat{y}_i = y_i), \qquad (1)$$

where N denotes the number of test instances, $\hat{y}_i$ the predicted label, y_i the ground-truth label, and $\mathbb{1}(\cdot)$ the indicator function.

3.3 Environment

All experiments were conducted on a workstation equipped with an NVIDIA RTX PRO 6000 (Blackwell generation) featuring 96 GB of VRAM. Model inference was performed using Ollama (v0.17.0).

3.4 Model Selection

We employed Gemma 3 (27B) (Team et al., 2025) for our evaluation. This choice is motivated by the model's high performance-to-resource ratio; specifically, its architecture allows for local execution on consumer-grade hardware with 24 GB of VRAM. This ensures that our approach remains computationally accessible.

4 Results

The results on the development set are presented in Appendix A. The most effective approach was the structured template combined with the standard prompting strategy in the zero-shot setting. At submission time, we submitted results from the first of the three seeds, achieving an accuracy of 67.75%.

Prompting Templates A thorough examination of the three prompting templates showed that the structured template is the strongest variant. Combining this template with zero-shot prompting yielded optimal outcomes for almost all prompting approaches. We observed the largest relative gains under zero-shot conditions. This pattern may be due to the increased importance of the base prompt formulation when no supplementary examples are provided.

Pairwise Comparison with the Best Performance The Pairwise Comparison approach yielded the highest accuracy on the test dataset. This finding suggests that evaluating both candidates jointly within a single prompt can introduce interference, whereas decomposed comparisons improve decision clarity. A salient observation is that the tie rate between the two pairwise similarity scores was approximately 30%.

Effect of Few-Shot Examples Few-shot examples yielded only marginal accuracy gains overall, while effects varied substantially across prompting

Rank	Team	Acc. (%)
1	COGNAC	78.00
2	FactUEP	75.75
3	AI-Monitors	75.00
4	TeleAI	74.75
5	YNU-HPCC	74.25
6	UTD-HLTRI	74.00
7	JCT	73.75
8	CuriosAI	73.50
9	Yam	73.00
10	CascadeMind	72.75
11	Team CV	70.75
12	hermeneutic_hools	70.50
13	NCL&HKU-NarrSim	70.25
13	NarSiL	70.25
15	DUTIR	69.75
16	CophiWue	69.50
17	CITD@UIT	69.25
17	ttda704	69.25
19	Narrative Nexus	68.50
20	harapalb	68.25
20	Comhis	68.25
22	SoloSemantics	68.00
23	**schmerle**	**67.75**
24	L3IRIT	65.75
25	NLP-FSDM	65.50
26	Team UBSE	65.25
27	MarSan	65.00
27	MoodMetric	65.00
29	PEU Lab	64.50
30	Narrative Team	64.25
31	ChulaNLP	63.50
32	blue	62.50
33	Team HausaNLP	61.50
34	TFB	61.25
35	Spinfo Cologne	60.25
36	LIAAD INESCTEC	59.75
37	Cryptix	59.50
38	CICL26	59.00
39	Duluth	58.50
40	IIITH Boys	57.75
41	Lacuna Inc.	57.25
42	Mendel292	56.50
43	PLlama	55.50
44	VerbaNexAI	53.50

Table 4: **Track A leaderboard.** Our team, **schmerle**, is highlighted.

approaches. For the default prompting approach, a small number of demonstrations (approximately six few-shot examples) proved most effective, while larger shot counts led to a gradual decline in accuracy, a pattern consistent with context overloading. The summarization approach tolerated a higher optimal shot count (approximately 12), plausibly because prior compression reduced effective context length and left sufficient capacity for additional examples. In contrast, the keyword filtering approach showed a monotonic decrease in accuracy with increasing shot count, suggesting that the high level of abstraction introduced during keyword extraction degraded the informativeness of in-context demonstrations.

Leaderboard As shown in Table 4, our system ranks 23th among 44 participating teams. Given that our submitted zero-shot structured template approach requires no task-specific annotation and no model fine-tuning, this placement is noteworthy. The 11% absolute accuracy gap to the top-ranked system indicates room for improvement.

5 Conclusion

We presented a training-free, prompting-only framework for narrative similarity detection under the SemEval-2026 Task 4 (Track A) setting. By systematically evaluating three prompt templates across five structural prompting strategies on Gemma 3 (27B), we demonstrated that carefully engineered prompts alone can yield competitive performance without task-specific fine-tuning or manual annotation. Our best configuration, a structured prompt template combined with pairwise 1-vs-1 story comparison, achieved 72.50% accuracy on the test set. On the leaderboard, the submitted zero-shot structured template approach achieved 67.75% accuracy. These results establish prompt engineering as a viable and reproducible baseline for narrative similarity tasks and highlight decomposed pairwise comparison as a particularly effective strategy for reducing attentional interference in multi-story reasoning.

Future work should explore several promising directions. First, integrating retrieval-augmented few-shot selection, replacing random sampling with semantically similar demonstrations, may yield more informative in-context examples. Second, chain-of-thought prompting and self-consistency decoding could be explored. Third, investigating larger LLMs and reasoning LLMs under the same prompting framework would clarify the extent to which our findings generalize across model families.

6 Limitations

Several limitations of the present work merit acknowledgment. First, while we evaluated a broad set of prompting approaches, all experiments were conducted with a single model, Gemma 3 (27B), limiting the generalizability of our conclusions to other model families and scales. Second, our few-shot examples were drawn randomly from a small sample split of only 39 instances, which constrains both example diversity and the reliability of shot-count comparisons. Finally, while our approach requires no annotation, it incurs non-trivial infer-

ence costs due to the multi-prompt structure of the pairwise and aspect-splitting configurations, which should be considered when deploying at scale.

References

Roland Barthes and Lionel Duisit. 1975. An introduction to the structural analysis of narrative. *New Literary History*, 6(2):237–272.

Regina Barzilay and Mirella Lapata. 2005. Modeling local coherence: An entity-based approach. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 141–148, Ann Arbor, Michigan. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Nathanael Chambers and Dan Jurafsky. 2008. Unsupervised learning of narrative event chains. In *Proceedings of ACL-08: HLT*, pages 789–797, Columbus, Ohio. Association for Computational Linguistics.

Jon Chun. 2024. AIStorySimilarity: Quantifying story similarity using narrative for search, IP infringement, and guided creativity. In *Proceedings of the 28th Conference on Computational Natural Language Learning*, pages 161–177, Miami, FL, USA. Association for Computational Linguistics.

Washington Cunha, Leonardo Rocha, and Marcos André Gonçalves. 2025. A thorough benchmark of automatic text classification: From traditional approaches to large language models. *Preprint*, arXiv:2504.01930.

David Elson and Kathleen McKeown. 2010. Automatic attribution of quoted speech in literary narrative. *Proceedings of the AAAI Conference on Artificial Intelligence*, 24(1):1013–1019.

Gérard Genette. 1980. *Narrative Discourse: An Essay in Method*. Cornell University Press, Ithaca, NY. Foreword by Jonathan Culler.

Hans Ole Hatzel, Ekaterina Artemova, Haimo Stiemer, Evelyn Gius, and Chris Biemann. 2026. SemEval-2025 task 4: Narrative similarity and narrative representation learning. In *Proceedings of the 20th International Workshop on Semantic Evaluation (SemEval-2026)*, San Diego, CA, USA.

Jia He, Mukund Rungta, David Koleczek, Arshdeep Sekhon, Franklin X Wang, and Sadid Hasan. 2024. Does prompt formatting have any impact on llm performance? *Preprint*, arXiv:2411.10541.

Nikitas . Karanikolas, Eirini Manga, Nikoletta Samaridi, Vaios Stergiopoulos, Eleni Tousidou, and Michael Vassilakopoulos. 2025. Strengths and weaknesses of llm-based and rule-based nlp technologies and their potential synergies. *Electronics*, 14(15).

Wendy G. Lehnert. 1981. Plot units and narrative summarization. *Cognitive Science*, 5(4):293–331.

Claude Lévi-Strauss. 1955. The structural study of myth. *The Journal of American Folklore*, 68(270):428–444.

Yuetian Mao, Junjie He, and Chunyang Chen. 2025. From prompts to templates: A systematic prompt template analysis for real-world llmapps. In *Proceedings of the 33rd ACM International Conference on the Foundations of Software Engineering*, FSE Companion '25, page 75–86, New York, NY, USA. Association for Computing Machinery.

Sania Nayab, Giulio Rossolini, Giorgio Buttazzo, Nicolamaria Manes, and Fabrizio Giacomelli. 2024. Concise thoughts: Impact of output length on llm reasoning and cost. *ArXiv*, abs/2407.19825.

Zabir Al Nazi, Md. Rajib Hossain, and Faisal Al Mamun. 2025. Evaluation of open and closed-source llms for low-resource language with zero-shot, few-shot, and chain-of-thought prompting. *Natural Language Processing Journal*, 10:100124.

Ahmed Hamza Osman, Naomie Salim, Mohammed Salem Binwahlan, Rihab Alteeb, and Albaraa Abuobieda. 2012. An improved plagiarism detection scheme based on semantic role labeling. *Applied Soft Computing*, 12(5):1493–1502.

Tanzir Pial and Steven Skiena. 2023. GNAT: A general narrative alignment tool. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14636–14652, Singapore. Association for Computational Linguistics.

Karl Pichotta and Raymond Mooney. 2016. Learning statistical scripts with lstm recurrent neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 30(1).

Vladimir Propp. 1968. *Morphology of the Folk Tale*, 2nd edition, volume 10 of *Bibliographical and Special Series*. University of Texas Press, Austin. Translated by Laurence Scott.

Seyed Mahmoud Sajjadi Mohammadabadi, Burak Cem Kara, Can Eyupoglu, Can Uzay, Mehmet Serkan Tosun, and Oktay Karakuş. 2025. A survey of large language models: Evolution, architectures, adaptation, benchmarking, applications, challenges, and societal implications. *Electronics*, 14(18).

Roger C. Schank and Robert P. Abelson. 1977. *Scripts, Plans, Goals, and Understanding: An Inquiry into Human Knowledge Structures*. The Artificial Intelligence Series. Lawrence Erlbaum Associates, Hillsdale, NJ.

Hongjin Su, Jungo Kasai, Chen Henry Wu, Weijia Shi, Tianlu Wang, Jiayi Xin, Rui Zhang, Mari Ostendorf, Luke Zettlemoyer, Noah A Smith, and Tao Yu. Selective annotation makes language models better few-shot learners.

Madhumita Sushil, Travis Zack, Divneet Mandair, Zhiwei Zheng, Ahmed Wali, Yan-Ning Yu, Yuwei Quan, Dmytro Lituiev, and Atul J Butte. 2024. A comparative study of large language model-based zero-shot inference and task-specific supervised classification of breast cancer pathology reports. *Journal of the American Medical Informatics Association*, 31(10):2315–2327.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, Louis Rouillard, Thomas Mesnard, Geoffrey Cideron, Jean bastien Grill, Sabela Ramos, Edouard Yvinec, Michelle Casbon, Etienne Pot, Ivo Penchev, and 197 others. 2025. Gemma 3 technical report. *Preprint*, arXiv:2503.19786.

Tzvetan Todorov. 1981. *Introduction to Poetics*, volume 1 of *Theory and History of Literature*. University of Minnesota Press, Minneapolis.

Bibek Upadhayay, Vahid Behzadan, and Amin Karbasi. 2024. Cognitive overload attack:prompt injection for long context. *Preprint*, arXiv:2410.11272.

Ming Wang, Yuanzhong Liu, Xiaoyu Liang, Songlian Li, Yijie Huang, Xiaoming Zhang, Sijia Shen, Chaofeng Guan, Daling Wang, Shi Feng, Huaiwen Zhang, Yifei Zhang, Minghui Zheng, and Chi Zhang. 2024. Langgpt: Rethinking structured reusable prompt design framework for llms from the programming language. *Preprint*, arXiv:2402.16929.

Jules White, Quchen Fu, Sam Hays, Michael Sandborn, Carlos Olea, Henry Gilbert, Ashraf Elnashar, Jesse Spencer-Smith, and Douglas C. Schmidt. 2023. A prompt pattern catalog to enhance prompt engineering with chatgpt. In *Proceedings of the 30th Conference on Pattern Languages of Programs*, PLoP '23, USA. The Hillside Group.

Zheng Zhang, Jinyi Li, Yihuai Lan, Xiang Wang, and Hao Wang. 2025. An empirical study on prompt compression for large language models. *Preprint*, arXiv:2505.00019.

Lixing Zhu, Runcong Zhao, Lin Gui, and Yulan He. 2023. Are NLP models good at tracing thoughts: An overview of narrative understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10098–10121, Singapore. Association for Computational Linguistics.

A Performance on the Development Set

	Identical Paper Prompt Copy				**Structured Template**				**Cognitive Load Reduction**			
	# Few-Shot Examples				*# Few-Shot Examples*				*# Few-Shot Examples*			
Task	*0*	*6*	*12*	*18*	*0*	*6*	*12*	*18*	*0*	*6*	*12*	*18*
prompting	58.50	65.67	63.17	65.17	**71.00**	69.33	66.00	66.50	63.00	67.50	63.33	64.33
summarize	57.00	58.17	58.67	59.50	**61.50**	60.17	59.67	60.67	61.00	61.00	59.17	59.50
filtering	55.00	**61.67**	58.50	61.33	57.50	59.00	57.17	58.83	55.00	59.00	58.00	57.67
one$_t o_o ne_l inear$	64.50	-	-	-	**67.00**	-	-	-	59.50	-	-	-
aspect$_s plitting$	54.50	-	-	-	**67.50**	-	-	-	59.00	-	-	-
Aggregate	57.90	61.84	60.11	62.00	**64.90**	62.83	60.95	62.00	59.50	62.50	60.17	60.50

Table 5: Development-set performance comparison across three prompting strategies (*Identical Paper Prompt Copy*, *Structured Template*, *Cognitive Load Reduction*) and four few-shot configurations. The last row reports aggregate results across all tasks. **Bold** values indicate the best performance per task and shot count.

B Prompt Template: Identical Paper Copy

```
## Task Description

- You are tasked with identifying similar stories.
- You will be presented with three stories, an **anchor-story**, and two choices, **story-a** and **story-b**.
- You are to determine which of the candidate stories, **story-a** and **story-b**, is the most similar to the **anchor-story**.
- The similarity between **story-a** and **story-b** is irrelevant.
- You select the candidate story that is more similar to the **anchor-story**.
- Specifically, you will consider the stories' narrative similarity.

## Narrative Similarity

The narrative similarity of stories can be broken down into three core aspects:

- (1) the abstract themes of the story,
- (2) the course of action
- (3) the story outcomes.
  At one extreme, this means that the story deals with the same themes and tells the same order of events with an identical outcome or
conclusion, just using a different wording; at the other extreme, the story might be completely different and lack any basis for comparison.

### We define these three aspects as follows:

- Abstract Theme describes the defining constellation of problems, central ideas, and core motifs of a story. The definition does not cover
the concrete setting of a story.
- Course of Action describes sequences of events, actions, conflicts, and turning points in a story and the order in which they happen.
- Outcomes describe the results of the plot at the end of the text, for example, the conflict resolution, the characters' fates, moral
lessons, etc. It does not cover intermediate statuses that change later in the story.

### Each aspect can take different forms in an actual pair of stories. Below, we list one example for each aspect:

#### The general setting of the story, if it strongly influences the events in the story or the events necessitate a specific setting
(abstract themes)

- A: On the week-long journey from Europe to the Americas, the crew members get into a heated conflict about the best ration packages.
- B: The flight to Mars is long. After several weeks, the astronauts become better friends than ever before, having to share the limited
resources.
- A and B share some similarities in that the polar opposite outcomes are both enabled by being cut off from the outside world.

#### The order of events in the story (course of action)

- A: After the ship capsizes and Alice barely makes it out alive, she starts living life to the fullest.
- B: Alice is living life to the fullest until, one day, her ship capsizes. She barely makes it out alive.
- A and B are similar in that both tell of a good life and a shipwreck (abstract theme), but they differ in the course of action, and the
order is very different.

#### The outcomes of events (story outcomes)

- A: The man intentionally drops a cup; it breaks.
- B: He accidentally swipes the bottle off the table, and it shatters.
- A and B are similar in that the events are comparable and lead to similar outcomes.

### There is a range of factors that expressly do NOT contribute to the narrative similarity:

- The style of writing in a story
- The concrete setting of a story (also including the time period).
- The names of the characters and locations
- The length of a text
- The level of detail in which the events are told

## Differentiating Between Similarity Aspects

Distinguishing the three aspects can be challenging. In general, it is important to consider each aspect independently. Often, pairs of
stories that are similar in terms of course of action will also share an abstract theme. However, it is possible that similar events emerge
from completely different surrounding circumstances. Outcomes, on the other hand, are clearly distinct from the other two aspects:
practically identical events in stories with comparable abstract themes can result in polar opposite outcomes.
When comparing abstract themes, it can help to explicitly formulate them.
Two stories share a general theme if there is a description that captures the defining circumstances of both stories.

## Output Format

Select **exactly one** answer:

- **{"text_a_is_closer": True}**
- **{"text_a_is_closer": False}**

> Do **not** provide explanations or additional text.
```

Figure 2: **Identical Paper Copy Prompt.** The prompt contains the task description, the definition of narrative similarity in the task context. And the output format for the model's structured output

C Prompt Template: Structured Template

```
## CONTEXT

Three stories are provided: an **Anchor** and two candidates (**Text A**, **Text B**).

Your task is to select **which candidate is most similar to the Anchor story**.

> Only compare each candidate to the Anchor. Ignore similarity between candidates.

## TASK

1. **Abstract Theme** - Compare core ideas and motifs, ignoring setting, names, and style.
2. **Course of Action** - Compare sequence, order, and structure of events and conflicts.
3. **Outcome** - Compare the final result, conflict resolution, and lesson conveyed.

> Consider each aspect independently. If only some align, weigh **the most central similarities**.

## NON-RELEVANT FACTORS

- Writing style or tone
- Time, location, or world type
- Character or place names
- Text length or detail level

## OUTPUT

Select **one option**:

- **{"text_a_is_closer": True}**
- **{"text_a_is_closer": False}**

> No explanations or extra text.
```

Figure 3: **Structured Template Prompt.** In a structured manner, this prompt comprises the context of the task, the task itself, any exceptions to the task, and the structured output format

D Prompt Template: Cognitive Load Reduction

```
## TASK DESCRIPTION

You have **3 stories**:

- Anchor story
- Text A
- Text B

**Objective:** Identify **which candidate story is more similar to the Anchor**.

## JUDGMENT CRITERIA

Compare **only Anchor → candidate** for **narrative similarity**, focusing on:

- **Abstract Theme**: Core ideas, motifs, and problems (ignore style, names, or setting)
- **Course of Action**: Event order, conflicts, and turning points
- **Outcome**: Final resolution or moral lesson (ignore intermediate states)

> Weigh aspects intuitively if only some match. Focus on main storyline, not side plots.

## IGNORE

- Style or tone
- Names, time, or location
- Text length
- Level of detail

## OUTPUT

Choose **one**:

- **{"text_a_is_closer": True}**
- **{"text_a_is_closer": False}**

> Do **not** add explanations.
```

Figure 4: **Cognitive Load Reduction Prompt.** This prompt contains a concise description and evaluation criteria for the narrative similarity task. It also delineates exceptions and the structured output format