FactUEP at SemEval-2026 Task 4: Structured Narrative Similarity Scoring with Aspect Decomposition and Weak-Signal Gating

Marcin Sawiński
marcin.sawinski@ue.poznan.pl
Poznań University of Economics and Business
Poznań, Poland

Abstract

This paper presents approach to narrative similarity prediction for SemEval-2026 Task 4 Track A. We introduce an LLM-based system that operationalizes the three core dimensions—Abstract Theme, Course of Action, and Outcomes—via schema-constrained prompting to enforce structured outputs and alignment with the annotation protocol. The system proceeds in three stages: structured aspect decomposition and scoring, weak-signal gating for low-confidence cases, and a targeted LLM-based tiebreak. The final model achieved near-human performance and ranked second on the Track A leaderboard.

1 Introduction

Narrative similarity concerns the degree to which stories share abstract structural patterns, independent of surface details such as setting, named entities, or writing style. SemEval-2026 Task 4 formalizes this problem as a comparative judgment task, requiring systems to determine which of two candidate stories is narratively closer to an anchor based on three dimensions: Abstract Theme, Course of Action, and Outcomes. The task is inherently subjective and characterized by low inter-annotator agreement, making robust operationalization challenging. In this work, we investigate whether structured decomposition, constrained prompting, and confidence-aware revision can improve large language model performance on this setting. We propose a schema-driven, multi-stage architecture that explicitly models aspect-level signals and conditionally applies revision via weak-signal gating.

This study addresses the following research questions:

RQ1: Does narrative similarity prediction require large-scale LLMs, or can smaller models achieve competitive performance?

RQ2: Does extended reasoning via verbose outputs improve performance?

RQ3: Can system confidence be estimated reliably and used to conditionally steer response generation?

RQ4: Does a multi-step prompting pipeline, including previous answer review or self-critique, improve prediction accuracy?

RQ5: Does decontextualization of input texts (i.e., removal of surface-specific details) improve narrative similarity assessment?

2 Background

The task comprises two tracks. Track A requires selecting, for each triple (anchor, A, B), the candidate narratively closer to the anchor. Track B requires generating embeddings whose distances reflect these similarity judgments (Hatzel et al., 2026).

Narrative similarity refers to abstract structural relatedness between stories, explicitly disregarding surface details. It is defined along three dimensions: Abstract Theme, Course of Action, and Outcomes.[1]

The task is not grounded in a specific narrative theory and provides no explicit weighting of the three aspects, aiming instead to reflect intuitive human judgments in an inherently subjective setting. This subjectivity is reflected in low inter-annotator agreement: Krippendorff's alpha was 0.33 for overall similarity and near chance for individual aspects (0.05, 0.07, 0.11 for theme, course of action, and outcomes).

Importantly, this definition differs from SemEval-2025 Task 10, which focused on ideological framing and public discourse rather than broader storytelling patterns (Piskorski et al., 2025). The approach to 2025 task 10 subtask 2 Narrative Classification was dominated by fine-tuning smaller models like Llama3.2 (Singh

[1] https://narrative-similarity-task.github.io/static/narrative-similarity_annotation-guidelines.pdf

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 1075–1088

et al., 2025) or Phi-4 and Qwen2.5 (Sun et al., 2025) and a multi-agent approach (Eljadiri and Nurbakova, 2025).

3 Dataset

The core dataset consists of annotated triples of story summaries drawn from Wikipedia. The development set includes 200 labeled triples, the test set - 400 triples.To enable meaningful human annotation, the authors employed the narrative embedding model *story-emb* (Hatzel and Biemann, 2024) to construct triples expected to share a certain degree of similarity. The organizers released additional 1,900 synthetic training triples generated by LLMs, however, their utility for training was limited, as the manually annotated dataset was significantly more challenging.

4 Methodology

4.1 Model selection

The first stage benchmarked a baseline prompt across multiple LLMs of varying parameter scales to assess the impact of model capacity on narrative similarity performance. Using identical prompting conditions, we compared different architectures and instruction-tuning regimes, namely two 8B models (Llama 3.1, DeepSeek-R1), GPT-OSS 20B and 120B and GPT-5.2

4.2 Prompt and context engineering

The second stage focused on designing multiple prompt variants that differed in the level of structural detail required and in the sequencing of intermediate reasoning steps.

Prompts were designed to enforce structured, schema-compliant outputs in a predefined JSON format. Each field imposed explicit constraints on content and granularity e.g.theme descriptions were required to be high-level abstractions, course-of-action representations had to be expressed as ordered event sequences, and outcomes were restricted to final resolution states only, decision output constrained to a single binary choice. In some variants, intermediate aspect scores were required before aggregation, ensuring that similarity judgments were decomposed prior to final comparison.

This schema-constrained design reduced reasoning drift, limited stylistic confounds, and increased consistency across model outputs.

The prompts were generated with GPT-5.2 under manual guidance on structure and were informed by the official annotation guidelines and aspect definitions of the task. The variants included the following prompt configurations.

Basic decomposition. We used prompt that enforced explicit decomposition of narrative similarity into three aspects before issuing a final judgment. This was implemented via schema-constrained response format with fixed fields per aspect. For both candidates, the model produced structured aspect-level comparisons followed by a final decision (see listing A.1).

Abstract. A set of four prompts designed to extract structured narrative representations of each aspect and the overall evaluation. The output format included list of abstract items present in the story, each item expressed in 2–5 words (see A.2).

Scoring. Prompts that combined structured aspect extraction with verbose justification and explicit numerical similarity scoring and three-class similarity classification (A, B, `NONE`). These variants required the model to compare aspect-level representations and assign similarity scores prior to producing a final decision (see A.3).

Answer Review. Prompts designed to re-evaluate prior predictions, particularly borderline cases. These prompts explicitly revisited earlier outputs to identify potential inconsistencies between aspect-level comparisons and the final verdict (see A.4).

4.3 Weak-Signal Gating

We introduced an additional processing stage to determine when answer revision was warranted. Replacing the final LLM verdict with hard or weighted soft voting over per-aspect decisions did not improve performance. Instead, we adopted a targeted strategy that preserved the original prediction in strong cases and applied revision only to low-confidence or internally inconsistent outputs. This was implemented via weak-gate conditions that selectively triggered an *Answer Review* stage. We define a weak-signal regime based on aspect-level similarity margins and decision consistency.

Let m_T, m_C, and m_O denote the per-aspect similarity margins between candidate A and B for theme, course of action, and outcomes, respectively:

$$m_T = \mathrm{sim}_T(A) - \mathrm{sim}_T(B), \quad (1)$$

$$m_C = \mathrm{sim}_C(A) - \mathrm{sim}_C(B), \quad (2)$$

$$m_O = \mathrm{sim}_O(A) - \mathrm{sim}_O(B). \quad (3)$$

We compute a weighted aggregate similarity margin, with empirically determined weights w_T, w_C, and w_O reflecting the relative importance of each aspect:

$$S_{\text{margin}} = \frac{w_T m_T + w_C m_C + w_O m_O}{w_T + w_C + w_O}. \quad (4)$$

In addition, we define the strongest per-aspect signal as:

$$M_{\max} = \max\left(|m_T|, |m_C|, |m_O|\right). \quad (5)$$

We further count the number of aspect-level decisions labeled as NONE:

$$N_{\text{none}} = \sum_{k \in \{T,C,O\}} \mathbf{1}[\text{closer}_k = \texttt{NONE}]. \quad (6)$$

A prediction is classified as *weak* if at least one of the following conditions holds:

$$|S_{\text{margin}}| < \tau_S \quad \text{(low global margin)} \quad (7)$$

$$M_{\max} < \tau_A \quad \text{(no strong aspect signal)} \quad (8)$$

$$N_{\text{none}} \geq k_{\text{none}} \quad \text{(insufficient aspect evidence)}. \quad (9)$$

Here, τ_S controls the minimum required aggregate margin, τ_A enforces a minimum per-aspect confidence threshold, and k_{none} specifies how many undefined aspect decisions trigger the gate (if the NONE rule is enabled).

If any of the above criteria are satisfied, the instance is routed to the *Answer Review* stage. Otherwise, the original prediction is retained. All parameter values and aspect weights were determined via experimental tuning on the development set.

5 Results

All models were evaluated on the development set using *Basic decomposition* prompt template.The 8B models (Llama 3.1, DeepSeek-R1) achieved below 60% accuracy, GPT-OSS models (20B, 120B) exceeded 65%, and GPT-5.2 surpassed 70% (see Table 1).

Table 1: Baseline decomposition prompt performance across models on the development set.

Model	Accuracy
Llama 3.1 8B	58.0
DeepSeek R1 8B	58.0
GPT OSS 20B	65.5
GPT OSS 120B	67.0
GPT-5.2	**73.5**

Subsequent experiments focused primarily on GPT-5.2, given its superior baseline performance.

Among single-prompt configurations, the highest score (77.5% accuracy) was achieved with the *No Monologue* variant. This prompt employed an elaborate system message detailing the task definition and decomposition rules, while limiting the final output strictly to a single-character decision (A or B). Enforcing structured reasoning instructions and suppressing any free-form explanations yielded the strongest results (see Table 2).

The second-best single-prompt configuration, *Verbose Scoring* (76.5%), used very similar system message but required structured output consisting of per-aspect abstraction, numerical similarity scoring and free-form contributions and decision explanations in the output. Longer output reduced single-pass accuracy, but produced richer intermediate signals for post-processing.

Applying a second-stage *Answer Review* prompt to outputs of the *Verbose Scoring* increased overall accuracy to 78%. This revision step corrected a number of initial errors, and also overturned some previously correct predictions. The aggregate effect remained positive, despite local instability.

Applying a second-stage *Answer Review* only to examples selected by the *Weak Gating* mechanism yielded further improvements, raising accuracy to 79.5%. The gating configuration was determined via random search over the parameter space. Although multiple parameter combinations produced comparable results, consistent patterns emerged across runs. For the final submission, we selected $\tau_S = 0.18$ and $\tau_A = 0.25$, with weights 0.4, 0.3 and 0.3 for Theme, Course of Action and Outputs with NONE rule disabled (see Figure 1).

Under the selected configuration, 69 out of 200 instances (34.5%) were classified as weak and routed to the *Answer Review* stage. Within this subset, 14 prediction flips occurred, corresponding to a 20.3% flip rate among weak cases (7% overall). Importantly, these revisions led to substantial gains: accuracy on weak instances improved from 62.3% to 71.0% (+8.7% points). Strong instances were left unchanged, preserving their baseline accuracy (83.97%). Overall accuracy increased from

76.5% to 79.5% (+3.0% points). These results indicate that the weak-gate mechanism effectively concentrates revision effort on low-confidence cases, improving performance without degrading already confident predictions.

We also explored prompt variants aimed at reducing verbosity in the *Verbose Scoring* configuration while preserving intermediate signals for two-stage processing. These modifications were intended to limit reasoning drift without sacrificing aspect-level information. Specifically, the *No Why* and *No Contribution* variants removed explicit rationales from the output, while the *Reordered* variant required the model to produce the final verdict before generating explanations.

Contrary to expectations, all such modifications degraded performance. This suggests that the explanatory components may play a stabilizing role in structured reasoning, and that altering the ordering or suppressing justificatory signals can disrupt the internal coherence of the model's decision process.

We also tested a two-stage prompting setup in which the model first generated abstract representations of theme, course of action, and outcomes, and then assessed similarity based on them. This reduced accuracy by about five percentage points, likely due to information loss during abstraction.

On the test set (n=400), *Verbose Scoring* reached 74.25% accuracy, *Answer Review* 75.25%, and *Weak Gating* 75.75%—drops of 2.25, 2.75, and 3.75 percentage points relative to the development set. The largest decline for *Weak Gating* suggests that its thresholds may have been overfitted to the development distribution, although the relative ordering of variants is preserved across both splits, indicating that the benefits of structured prompting and confidence-aware revision generalize beyond the tuning sample. To assess whether the 1.5-point gain of *Weak Gating* over *Verbose Scoring* is statistically meaningful, we examined the 20 discordant pairs (13 helpful vs. 7 harmful flips). McNemar's exact test yielded p=0.26, and a paired bootstrap over 10,000 resamples produced a 95% CI of $[-0.75, +3.75]$ percentage points (p_{boot}=0.20). Neither test rejects the null, and the interval includes zero. The direction is consistent with the development set and with the underlying mechanism, but the test sample is too small to establish significance, so we interpret *Weak Gating* as directionally useful but not statistically conclusive.

Table 2: GPT-5.2 performance across prompt configurations on the development and test set.

	Accuracy	
Prompt Variant	development	test
Single-Prompt		
No Monologue	**77.50**	
Verbose Scoring	76.50	74.25
No why	75.00	
No contribution	74.00	
Reordered	74.50	
Two-Prompt		
Answer Review	78.00	75.25
Weak Gating	**79.50**	**75.75**

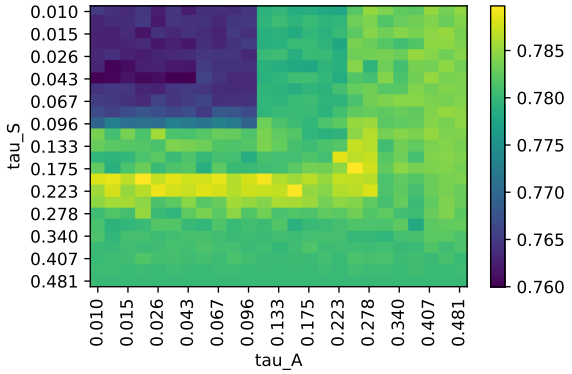

Figure 1: Accuracy heatmap over weak-gate thresholds τ_S and τ_A.

6 Discussion

RQ1: Does narrative similarity prediction require large-scale LLMs? Yes. Results indicate a clear model-scale and training regime dependency. Smaller 8B models operated only marginally above chance, while larger GPT-OSS models improved by +7.0 / +9.0 points with little difference between 20B and 120B models. GPT-5.2 consistently surpassed 70% across prompt variants, however, the gap between results suggests that architectural design and training regime, not parameter count alone, play a decisive role.

RQ2: Does extended reasoning via verbose outputs improve performance? No. Extended reasoning via verbose outputs did not improve single-pass accuracy. The *Verbose Scoring* prompt (76.5%) underperformed the more constrained *No Monologue* variant (77.5%), indicating that longer outputs can introduce reasoning drift. Thus, elaborate reasoning output does not automatically trans-

late into better performance.

RQ3: Can system confidence be estimated and used to conditionally steer response generation? Yes. *Weak Gating* effectively identified low-confidence cases (34.5% of instances) and selectively applied revision. This improved accuracy on weak cases from 62.3% to 71.0% without degrading strong predictions, yielding a +3.0% points (dev) and +1.5% point (test) gain. These results demonstrate that margin-based confidence estimation can reliably guide conditional response refinement.

RQ4: Does a multi-step prompting pipeline, including previous answer review or self-critique, improve prediction accuracy? Yes. Multi-step prompting improves performance when applied selectively. Unconditional *Answer Review* yielded modest gains (+1.5% point dev, +1.0% point test), while gated application achieved larger improvements (+3.0% points dev, +1.5% point test). However, decomposition-first pipelines that separated abstraction and judgment into separate steps reduced accuracy, indicating that multi-stage architectures are beneficial only when initial signals are preserved without excessive information loss.

RQ5: Does decontextualization of input texts (i.e., removal of surface-specific details) improve narrative similarity assessment? No. The *Abstract* prompt, which used only generated decontextualized representations of each aspect without presenting the full original text of stories, underperformed the full text setup by about -5% points. This suggests that surface details may provide important cues for similarity assessment, and that abstraction can lead to information loss detrimental to performance.

7 Conclusions and Future Work

Across experiments, three main findings emerge.

First, model scale and architecture matter. Smaller 8B models are not suited for the task. However, improvements were not strictly monotonic with parameter count, suggesting that alignment and training regime are at least as important as scale alone.

Second, explicit decomposition improves the score by better mimicking the annotator's thinking process and decisions. Highly constrained prompts that suppressed free-form explanations achieved the best standalone performance, indicating that limiting reasoning drift can outweigh the benefits of verbose deliberation. At the same time, structured intermediate signals (e.g., aspect-level scores) proved valuable when used for post-hoc calibration.

Third, confidence-aware revision is effective. The proposed Weak-Signal Gating mechanism reliably identified low-margin or internally inconsistent predictions and selectively routed them to a second-stage review. The results demonstrate that margin-based confidence estimation can be operationalized in a principled and computationally lightweight manner.

Limitations. Despite a strong leaderboard position, important limitations remain. First, the evaluation sample is small: on the 400-example dataset, a one percentage point gain equals only four additional correct predictions. Although patterns were consistent across dev and test sets, improvements of 1–3 percentage points remain vulnerable to statistical variance and should be interpreted cautiously.

Second, inter-annotator agreement on individual narrative aspects is low, reflecting the intrinsic subjectivity of narrative similarity judgments. The task formulation leaves aspect weighting unspecified, introducing variability that models must implicitly resolve.

Third, our approach relies on high-capacity LLMs, and its effectiveness may not transfer to resource-constrained settings without additional distillation or compression.

Research directions. First, incorporating more formal narrative theory into dataset generation and system design may improve interpretability, reduce ambiguity, and increase stability.

Second, extending experiments to substantially larger annotated datasets would allow more reliable estimation of effect sizes and statistical significance. Larger samples would reduce sensitivity to sampling noise and enable finer-grained analysis of gating thresholds, flip behavior, and multi-stage prompting strategies.

References

Mohamed Nour Eljadiri and Diana Nurbakova. 2025. Team INSALyon2 at SemEval-2025 task 10: A zero-shot agentic approach to text classification. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 965–980, Vienna, Austria. Association for Computational Linguistics.

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.

Hans Ole Hatzel and Chris Biemann. 2024. Story embeddings — narrative-focused representations of fictional stories. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5931–5943, Miami, Florida, USA. Association for Computational Linguistics.

Jakub Piskorski, Tarek Mahmoud, Nikolaos Nikolaidis, Ricardo Campos, Alipio Mario Jorge, Dimitar Dimitrov, Purificação Silvano, Roman Yangarber, Shivam Sharma, Tanmoy Chakraborty, Nuno Guimaraes, Elisa Sartori, Nicolas Stefanovitch, Zhuohan Xie, Preslav Nakov, and Giovanni Da San Martino. 2025. SemEval 2025 task 10: Multilingual characterization and extraction of narratives from online news. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 2610–2643, Vienna, Austria. Association for Computational Linguistics.

Iknoor Singh, Carolina Scarton, and Kalina Bontcheva. 2025. GateNLP at SemEval-2025 task 10: Hierarchical three-step prompting for multilingual narrative classification. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 148–154, Vienna, Austria. Association for Computational Linguistics.

Ling Sun, Xue Wan, Yuyang Lin, Fengping Su, and Pengfei Chen. 2025. PATeam at SemEval-2025 task 10: Two-stage news analytical framework: Target-oriented semantic segmentation and sequence generation LLMs for cross-lingual entity and narrative analysis. In *Proceedings of the 19th International Workshop on Semantic Evaluation (SemEval-2025)*, pages 2121–2132, Vienna, Austria. Association for Computational Linguistics.

A Prompt templates

A.1 Basic decomposition

```
"messages": [
    {
        "role": "system",
        "content":
"You are an expert on stories and narratives.
Tell us which of two stories is narratively
similar to the anchor story. Decompose the story
into the three official dimensions:
Abstract Theme: The ideas and motives of the
story.
Course of Action: The sequence of central events,
turning points, etc.
Outcomes: The results of a story."
    },
    {
        "role": "user",
        "content":
"Anchor story: {{anchor_text}}
Story A: {{text_a}}
Story B: {{text_b}}"
    }
]

"response_format": {
    "model_name": "SimilarityPrediction",
    "enums": {
        "ResponseEnum": [
            "A",
            "B"
        ]
    },
    "fields": [
        // --- Themes ---
        {
            "name": "anchor_story_theme",
            "type": "str",
            "description": "Abstract Theme of
            Anchor Story."
        },
        {
            "name": "story_a_theme",
            "type": "str",
            "description": "Abstract Theme of
            Story A."
        },
        {
            "name": "story_b_theme",
            "type": "str",
            "description": "Abstract Theme of
            Story B."
        },
        {
            "name": "closer_by_theme",
            "type": "enum",
            "enum_name": "ResponseEnum",
            "description": "Which story is closer
            in terms of Abstract Theme"
        },
        // --- Course ---
        {
            "name": "anchor_story_course",
            "type": "str",
            "description": "Course of Action of
            Anchor Story."
        },
        {
            "name": "story_a_course",
            "type": "str",
            "description": "Course of Action of
            Story A."
        },
        {
            "name": "story_b_course",
            "type": "str",
            "description": "Course of Action of
            Story B."
        },
        {
            "name": "closer_by_course",
            "type": "enum",
            "enum_name": "ResponseEnum",
            "description": "Which story is closer
            in terms of Course of Action"
        },
        // --- Outcomes ---
        {
            "name": "anchor_story_outcomes",
            "type": "str",
            "description": "Outcomes of Anchor
            Story."
```

```
        },
        {
            "name": "story_a_outcomes",
            "type": "str",
            "description": "Outcomes of Story A."
        },
        {
            "name": "story_b_outcomes",
            "type": "str",
            "description": "Outcomes of Story B."
        },
        {
            "name": "closer_by_outcomes",
            "type": "enum",
            "enum_name": "ResponseEnum",
            "description": "Which story is closer
            in terms of Outcomes"
        },
        // --- Explanation + final ---
        {
            "name": "explanation",
            "type": "str",
            "description": "Short rationale which
            story is more similar overall"
        },
        {
            "name": "closer",
            "type": "enum",
            "enum_name": "ResponseEnum",
            "description": "Which story is closer
            overall."
        }
    ]
}
```

A.2 Abstract

A.2.1 Theme abstraction

```
"messages": [
    {
        "role": "system",
        "content":
"You are an expert on identifying narratively
similar stories. 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, and (3) the story
outcomes. Extract Abstract Theme of the story
that describes the defining constellation of
problems, central ideas, and core motifs of a
story. Focus on the core aspects of stories,
potentially largely ignoring side storylines.
Consider Abstract Theme as the general setting of
the story, if it strongly influences the events
in the story or the events necessitate a specific
setting.
Two example stories:
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.
Both A and B share Abstract Theme that the polar
opposite outcomes are both enabled by being cut
off from the outside world.
Ignore the concrete setting of a story (also
including the time period) and the names of the
characters and locations.  If more then one
Abstract Theme is present describe them as a
list. Each Abstract Theme should be briefly
characterized in 2-5 words.
        "
    },
    {
        "role": "user",
        "content": "Story: {{text}}"
    }
]

"response_format": {
    "model_name": "AbstractTheme",
    "fields": [
        {
            "name": "abstract_theme",
            "type": "list",
            "item_type": "str",
            "description": "List of abstract
            themes present in the story, each
            described in in 2-5 words."
        }
    ]
}
```

A.2.2 Course of action abstraction

```
"messages": [
    {
        "role": "system",
        "content":
"You are an expert on identifying narratively
similar stories. 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, and (3) the story
outcomes. Extract Course of Action that describes
sequences of events, actions, conflicts, and
turning points in a story and the order in which
they happen. Focus on the core aspects of stories,
potentially largely ignoring side storylines.
Consider Course of Action as the order of events,
that build the story. Two example stories:
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.

Course of Action of Story A is ['Sudden
catastrophic accident', Protagonist narrowly
survives', 'Personal perspective shifts','Life
lived fully'].
Course of Action of Story B is reversed ['Life
lived fully', 'Sudden catastrophic accident',
Protagonist narrowly survives'].
Ignore the concrete setting of a story (also
including the time period) and the names of the
characters and locations.  If more then one
Abstract Theme is present describe them as a
list. Each Abstract Theme should be briefly
characterized in 2-5 words."
    },
    {
        "role": "user",
        "content": "Story: {{text}}"
    }
```

```
]

"response_format": {
    "model_name": "CourseOfAction",
    "fields": [
        {
            "name": "course_of_action",
            "type": "list",
            "item_type": "str",
            "description": "Ordered list of
            abstract events present in the story,
            each described in in 2-5 words."
        }
    ]
}
```

A.2.3 Outcomes abstraction

```
"messages": [
    {
        "role": "system",
        "content":
"You are an expert on identifying narratively
similar stories. 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, and (3) the story
outcomes. Extract Outcomes that 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. Focus on the core aspects of
stories, potentially largely ignoring side
storylines. Two example stories:
A: The man intentionally drops a cup; it breaks.
B: He accidentally swipes the bottle off the
table, and it shatters.
Both A and B share Outcome that object is broken.
Ignore the concrete setting of a story (also
including the time period) and the names of the
characters and locations.  If more then one
Outcome is present describe them as a list. Each
Abstract Action should be briefly characterized
in 2-5 words."
    },
    {
        "role": "user",
        "content": "Story: {{text}}"
    }
]

"response_format": {
    "model_name": "Outcome",
    "fields": [
        {
            "name": "outcome",
            "type": "list",
            "item_type": "str",
            "description": "List of outcomes
            present in the story, each described
            in in 2-5 words."
        }
    ]
}
```

A.2.4 Score abstraction

```
"messages": [
    {
        "role": "system",
        "content":
"You are an expert on stories and narratives.
Tell us which of two stories is narratively
similar to the anchor story.
The story is decomposed into the three
dimensions:
Abstract Theme: The ideas and motives of the
story.
Course of Action: The sequence of central events,
turning points, etc.
Outcomes: The results of a story."
    },
    {
        "role": "user",
        "content":
"Anchor story Abstract theme:
{{anchor_abstract_theme}}
Story A Abstract theme:
{{story_a_abstract_theme}}
Story B Abstract theme:
{{story_b_abstract_theme}}
Anchor story Course of Action:
{{anchor_course_of_action}}
Story A Course of Action:
{{story_a_course_of_action}}
Story B Course of Action:
{{story_b_course_of_action}}
Anchor story Outcomes: {{anchor_outcomes}}
Story A Outcomes: {{story_a_outcomes}}
Story B Outcomes: {{story_b_outcomes}}"
    }
]

"response_format": {
    "model_name": "SimilarityPrediction",
    "enums": {
        "ResponseEnum": [
            "A",
            "B"
        ]
    },
    "fields": [
        {
            "name": "closer_by_theme",
            "type": "enum",
            "enum_name": "ResponseEnum",
            "description": "Which story is closer
            in terms of Abstract Theme"
        },
        {
            "name": "closer_by_course",
            "type": "enum",
            "enum_name": "ResponseEnum",
            "description": "Which story is closer
            in terms of Course of Action"
        },
        {
            "name": "closer_by_outcomes",
            "type": "enum",
            "enum_name": "ResponseEnum",
            "description": "Which story is closer
            in terms of Outcomes"
        },
        {
            "name": "explanation",
            "type": "str",
            "description": "Short rationale which
            story is more similar overall"
        },
        {
            "name": "closer",
            "type": "enum",
```

```
            "enum_name": "ResponseEnum",
            "description": "Which story is closer
            overall."
        }
    ]
}
```

A.3 Scoring

A.3.1 No Monologue

```
"messages": [
    {
        "role": "system",
        "content":
"You are an expert evaluator of narrative
similarity. Task:Decide which candidate story (A
or B) is narratively closer to the Anchor story.
Compare each candidate ONLY to the Anchor.
Similarity between A and B is irrelevant.
Narrative similarity is defined by three aspects:
1) Abstract Theme
- Core ideas, problems, motives, or tensions that
define the story.
- High-level and generalized.
- Do NOT include names, specific locations, time
periods, or writing style.
2) Course of Action
- Ordered sequence of causally important events
and turning points.
- Include only events necessary for the story
logic.
- Event order matters.
3) Outcomes
- Final results or resolutions at the end of the
story.
- Do NOT include intermediate or temporary
states.
Instructions:
- Assess A vs Anchor and B vs Anchor
independently.
- Weigh the three aspects intuitively.
- A single decisive aspect may outweigh weaker
similarities in others.
- Ignore writing style, length, level of detail,
character names, and locations.
Output constraint:
- Respond with exactly ONE character: either 'A'
or 'B'.
- Do NOT explain your reasoning.
- Do NOT output anything else."
    },
    {
        "role": "user",
        "content":
"Anchor story: {{anchor_text}}
Story A: {{text_a}}
Story B: {{text_b}}"
    }
]

"response_format": {
    "model_name": "SimilarityPrediction",
    "enums": {
        "ResponseEnum": [
            "A",
            "B"
        ],
    },
    "fields": [
        {
            "name": "closer",
            "type": "enum",
            "enum_name": "ResponseEnum",
            "description": "Closer story (A or B
            only)."
        }
    ]
}
```

A.3.2 Verbose Scoring

```
"messages": [
    {
        "role": "system",
        "content":
"You are an expert evaluator of narrative
similarity. Task:Decide which candidate story (A
or B) is narratively closer to the Anchor story.
Compare each candidate ONLY to the Anchor.
Similarity between A and B is irrelevant.
Narrative similarity is defined by three aspects:

1) Abstract Theme
- Core ideas, problems, motives, or tensions that
define the story.
- High-level and generalized.
- Do NOT include names, specific locations, time
periods, or writing style.
2) Course of Action
- Ordered sequence of causally important events
and turning points.
- Include only events necessary for the story
logic.
- Event order matters.
3) Outcomes
- Final results or resolutions at the end of the
story.
- Do NOT include intermediate or temporary
states.
Non-contributing factors:Writing style, names,
locations, time period, text length, and level of
detail.
Procedure (follow strictly):
A) Decompose the Anchor, Story A, and Story B
into the three aspects above.
- Use short phrases or compact sentences.
- Be precise and minimal; no storytelling or
embellishment.
B) For EACH aspect separately:
- Assess similarity between Anchor and Story A,
and between Anchor and Story B.
- Provide a similarity score in [0.0, 1.0] for
each comparison.
- Decide which story is closer for that aspect:
A, B, or NONE if neither is meaningfully similar.
C) Overall decision:
- Choose the overall closer story (A or B only).
Weight aspects by their importance in this
specific case.
- A single decisive aspect may outweigh two weak
ones.
- If all similarities are weak, still choose the
best available overlap.
D) Aspect contribution:
- For the FINAL chosen story only, mark which
aspects significantly contribute.
- If an aspect contributes, provide a concise 2–5
word reason.
E) Consistency requirements:
- If an aspect is marked as contributing=True, it
must not be NONE for the chosen story.
- If both similarity scores for an aspect are
extremely low, prefer NONE.
```

```
Follow the response schema exactly. Do not add
extra fields or commentary."
    },
    {
        "role": "user",
        "content":
"Anchor story: {{anchor_text}}
Story A: {{text_a}}
Story B: {{text_b}}"
    }
]

"response_format": {
    "model_name": "SimilarityPrediction",
    "enums": {
        "ResponseEnum": [
            "A",
            "B"
        ],
        "AspectChoiceEnum": [
            "A",
            "B",
            "NONE"
        ]
    },
    "fields": [
        // --- Themes ---
        {
            "name": "anchor_theme",
            "type": "str",
            "description": "Abstract Theme of
            Anchor (high-level, generalized)."
        },
        {
            "name": "story_a_theme",
            "type": "str",
            "description": "Abstract Theme of
            Story A."
        },
        {
            "name": "story_b_theme",
            "type": "str",
            "description": "Abstract Theme of
            Story B."
        },
        {
            "name": "theme_sim_a",
            "type": "float",
            "description": "Theme similarity
            between Anchor and A (0.0-1.0)."
        },
        {
            "name": "theme_sim_b",
            "type": "float",
            "description": "Theme similarity
            between Anchor and B (0.0-1.0)."
        },
        {
            "name": "closer_by_theme",
            "type": "enum",
            "enum_name": "AspectChoiceEnum",
            "description": "Closer by Abstract
            Theme."
        },
        // --- Course ---
        {
            "name": "anchor_course",
            "type": "str",
            "description": "Course of Action of
            Anchor (ordered, causal essentials)."
        },
        {
            "name": "story_a_course",
            "type": "str",
            "description": "Course of Action of
            Story A."
        },
        {
            "name": "story_b_course",
            "type": "str",
            "description": "Course of Action of
            Story B."
        },
        {
            "name": "course_sim_a",
            "type": "float",
            "description": "Course similarity
            between Anchor and A (0.0-1.0)."
        },
        {
            "name": "course_sim_b",
            "type": "float",
            "description": "Course similarity
            between Anchor and B (0.0-1.0)."
        },
        {
            "name": "closer_by_course",
            "type": "enum",
            "enum_name": "AspectChoiceEnum",
            "description": "Closer by Course of
            Action."
        },
        // --- Outcomes ---
        {
            "name": "anchor_outcomes",
            "type": "str",
            "description": "Final outcomes of
            Anchor (end state only)."
        },
        {
            "name": "story_a_outcomes",
            "type": "str",
            "description": "Final outcomes of
            Story A."
        },
        {
            "name": "story_b_outcomes",
            "type": "str",
            "description": "Final outcomes of
            Story B."
        },
        {
            "name": "outcomes_sim_a",
            "type": "float",
            "description": "Outcome similarity
            between Anchor and A (0.0-1.0)."
        },
        {
            "name": "outcomes_sim_b",
            "type": "float",
            "description": "Outcome similarity
            between Anchor and B (0.0-1.0)."
        },
        {
            "name": "closer_by_outcomes",
            "type": "enum",
            "enum_name": "AspectChoiceEnum",
            "description": "Closer by Outcomes."
        },
        // --- Regularizers ---
        {
            "name": "theme_contributes",
```

```
            "type": "bool",
            "description": "Theme significantly
            contributes to FINAL decision."
        },
        {
            "name": "theme_why_2to5",
            "type": "str",
            "description": "2-5 word reason if
            Theme contributes; else empty."
        },
        {
            "name": "course_contributes",
            "type": "bool",
            "description": "Course significantly
            contributes to FINAL decision."
        },
        {
            "name": "course_why_2to5",
            "type": "str",
            "description": "2-5 word reason if
            Course contributes; else empty."
        },
        {
            "name": "outcomes_contributes",
            "type": "bool",
           "description": "Outcomes significantly
            contribute to FINAL decision."
        },
        {
            "name": "outcomes_why_2to5",
            "type": "str",
            "description": "2-5 word reason if
            Outcomes contribute; else empty."
        },
        // --- Explanation + final ---
        {
            "name": "explanation",
            "type": "str",
            "description": "Brief justification
            naming which aspects dominated and
            why."
        },
        {
            "name": "closer",
            "type": "enum",
            "enum_name": "ResponseEnum",
            "description": "Overall closer story
            (A or B only)."
        }
    ]
}
```

A.3.3 No why

Note: Fields `messages` and `enums` are identical to the *Verbose Scoring* prompt, and only the `response_format` field controlling the output is constrained.

```
"response_format": {
"model_name": "SimilarityPrediction_NoWhy",
"fields": [
    // --- Themes ---
    {
        "name": "anchor_theme",
        "type": "str"
    },
    {
        "name": "story_a_theme",
        "type": "str"
    },
    {
        "name": "story_b_theme",
        "type": "str"
    },
    {
        "name": "theme_sim_a",
        "type": "float"
    },
    {
        "name": "theme_sim_b",
        "type": "float"
    },
    {
        "name": "closer_by_theme",
        "type": "enum",
        "enum_name": "AspectChoiceEnum"
    },
    // --- Course ---
    {
        "name": "anchor_course",
        "type": "str"
    },
    {
        "name": "story_a_course",
        "type": "str"
    },
    {
        "name": "story_b_course",
        "type": "str"
    },
    {
        "name": "course_sim_a",
        "type": "float"
    },
    {
        "name": "course_sim_b",
        "type": "float"
    },
    {
        "name": "closer_by_course",
        "type": "enum",
        "enum_name": "AspectChoiceEnum"
    },
    // --- Outcomes ---
    {
        "name": "anchor_outcomes",
        "type": "str"
    },
    {
        "name": "story_a_outcomes",
        "type": "str"
    },
    {
        "name": "story_b_outcomes",
        "type": "str"
    },
    {
        "name": "outcomes_sim_a",
        "type": "float"
    },
    {
        "name": "outcomes_sim_b",
        "type": "float"
    },
    {
        "name": "closer_by_outcomes",
        "type": "enum",
        "enum_name": "AspectChoiceEnum"
    },
    // --- Regularizers (BOOL ONLY) ---
    {
        "name": "theme_contributes",
```

```
        "type": "bool"
    },
    {
        "name": "course_contributes",
        "type": "bool"
    },
    {
        "name": "outcomes_contributes",
        "type": "bool"
    },
    // --- Explanation + final ---
    {
        "name": "explanation",
        "type": "str"
    },
    {
        "name": "closer",
        "type": "enum",
        "enum_name": "ResponseEnum"
    }
]
}
```

A.3.4 No contribution

Note: Fields `messages` and `enums` are identical to the *Verbose Scoring* prompt, and only the `response_format` field controlling the output is constrained.

```
"response_format": {
"model_name": "SimilarityPrediction_NoContrib",
"fields": [
    // --- Themes ---
    {
        "name": "anchor_theme",
        "type": "str"
    },
    {
        "name": "story_a_theme",
        "type": "str"
    },
    {
        "name": "story_b_theme",
        "type": "str"
    },
    {
        "name": "theme_sim_a",
        "type": "float"
    },
    {
        "name": "theme_sim_b",
        "type": "float"
    },
    {
        "name": "closer_by_theme",
        "type": "enum",
        "enum_name": "AspectChoiceEnum"
    },
    // --- Course ---
    {
        "name": "anchor_course",
        "type": "str"
    },
    {
        "name": "story_a_course",
        "type": "str"
    },
    {
        "name": "story_b_course",
        "type": "str"
    },
    {
        "name": "course_sim_a",
        "type": "float"
    },
    {
        "name": "course_sim_b",
        "type": "float"
    },
    {
        "name": "closer_by_course",
        "type": "enum",
        "enum_name": "AspectChoiceEnum"
    },
    // --- Outcomes ---
    {
        "name": "anchor_outcomes",
        "type": "str"
    },
    {
        "name": "story_a_outcomes",
        "type": "str"
    },
    {
        "name": "story_b_outcomes",
        "type": "str"
    },
    {
        "name": "outcomes_sim_a",
        "type": "float"
    },
    {
        "name": "outcomes_sim_b",
        "type": "float"
    },
    {
        "name": "closer_by_outcomes",
        "type": "enum",
        "enum_name": "AspectChoiceEnum"
    },
    // --- Explanation + final ---
    {
        "name": "explanation",
        "type": "str"
    },
    {
        "name": "closer",
        "type": "enum",
        "enum_name": "ResponseEnum"
    }
]
}
```

A.3.5 Reordered

Note: Fields `messages` and `enums` are identical to the *Verbose Scoring* prompt, and only the `response_format` field controlling the output is constrained.

```
"response_format": {
"model_name": "SimilarityPrediction_Reordered",
"fields": [
    // --- Themes ---
    {
        "name": "anchor_theme",
        "type": "str"
    },
    {
        "name": "story_a_theme",
```

```
    "type": "str"
},
{
    "name": "story_b_theme",
    "type": "str"
},
{
    "name": "theme_sim_a",
    "type": "float"
},
{
    "name": "theme_sim_b",
    "type": "float"
},
{
    "name": "closer_by_theme",
    "type": "enum",
    "enum_name": "AspectChoiceEnum"
},
// --- Course ---
{
    "name": "anchor_course",
    "type": "str"
},
{
    "name": "story_a_course",
    "type": "str"
},
{
    "name": "story_b_course",
    "type": "str"
},
{
    "name": "course_sim_a",
    "type": "float"
},
{
    "name": "course_sim_b",
    "type": "float"
},
{
    "name": "closer_by_course",
    "type": "enum",
    "enum_name": "AspectChoiceEnum"
},
// --- Outcomes --
{
    "name": "anchor_outcomes",
    "type": "str"
},
{
    "name": "story_a_outcomes",
    "type": "str"
},
{
    "name": "story_b_outcomes",
    "type": "str"
},
{
    "name": "outcomes_sim_a",
    "type": "float"
},
{
    "name": "outcomes_sim_b",
    "type": "float"
},
{
    "name": "closer_by_outcomes",
    "type": "enum",
    "enum_name": "AspectChoiceEnum"
},
// --- EARLY FINAL DECISION ---
    {
        "name": "closer",
        "type": "enum",
        "enum_name": "ResponseEnum"
    },
    // --- Regularizers ---
    {
        "name": "theme_contributes",
        "type": "bool"
    },
    {
        "name": "theme_why_2to5",
        "type": "str"
    },
    {
        "name": "course_contributes",
        "type": "bool"
    },
    {
        "name": "course_why_2to5",
        "type": "str"
    },
    {
        "name": "outcomes_contributes",
        "type": "bool"
    },
    {
        "name": "outcomes_why_2to5",
        "type": "str"
    },
    {
        "name": "explanation",
        "type": "str"
    }
]
}
```

A.4 Answer Review

```
"messages": [
    {
        "role": "system",
        "content":
"You are an expert evaluator of narrative
similarity.
You will receive:
- The Anchor story and two candidates (A, B)
- A prior model's decompositions and similarity
scores by aspect
Your job is NOT to re-decompose. Your job is ONLY
to correct borderline mistakes.
Decide which story (A or B) is narratively closer
to the Anchor.
Rules (follow strictly):
1) Do not reward broad genre/topic overlap (e.g.,
'crime', 'war', 'romance', 'politics') unless it
plays the same causal role.
2) Prefer the candidate that matches the Anchor's
narrative STRUCTURE:
- causal role pattern (who acts vs who is acted
upon)
- type of arc (investigation, escape/pursuit,
moral dilemma, downfall, redemption,
coming-of-age, etc.)
- resolution direction (improves vs worsens vs
unresolved)
3) Use the provided aspect decompositions and
scores as evidence, but you MAY override them if
they appear inconsistent with structure/outcome
direction. 4) Output A or B only. No NONE.
Follow the response schema exactly. Do not add
extra fields."
```

```
    },
    {
        "role": "user",
        "content":
"ANCHOR: {{anchor_text}}
STORY A: {{text_a}}
STORY B: {{text_b}}
---- PRIOR MODEL OUTPUT (EVIDENCE) ----
Anchor Theme: {{anchor_theme}} A Theme:
{{story_a_theme}} B Theme: {{story_b_theme}}
theme_sim_a={{theme_sim_a}},
theme_sim_b={{theme_sim_b}},
closer_by_theme={{closer_by_theme}}
Anchor Course: {{anchor_course}} A Course:
{{story_a_course}} B Course: {{story_b_course}}
course_sim_a={{course_sim_a}},
course_sim_b={{course_sim_b}},
closer_by_course={{closer_by_course}}
Anchor Outcomes: {{anchor_outcomes}} A Outcomes:
{{story_a_outcomes}} B Outcomes:
{{story_b_outcomes}}
outcomes_sim_a={{outcomes_sim_a}},
outcomes_sim_b={{outcomes_sim_b}},
closer_by_outcomes={{closer_by_outcomes}}
Prior overall closer: {{closer}} Prior
explanation: {{explanation}}"
    }
]
"response_format": {
    "model_name": "SimilarityTieBreak_Min",
    "enums": {
        "ResponseEnum": [
            "A",
            "B"
        ]
    },
    "fields": [
        {
            "name": "closer",
            "type": "enum",
            "enum_name": "ResponseEnum",
            "description": "Which story is
            narratively closer overall (A or B
            only)."
        },
        {
            "name": "why",
            "type": "str",
            "description": "One sentence naming
            the decisive structural reason (not
            genre)."
        }
    ]
}
```