JCT at SemEval-2026 Task 4: A Multi-Method Approach to Narrative Story Similarity

Dvori Rosenfeld, Rinat Walles and Chaya Liebeskind
Jerusalem College of Technology
21 Havaad Haleumi St., 91160
Jerusalem, Israel
{dvorilus, rinatwalles, liebchaya}@gmail.com

Abstract

Narrative similarity detection involves understanding the underlying structure of a story rather than just matching similar words or phrases. This paper details our multi-strategy approach to the SemEval-2026 Task on Narrative Similarity, which requires identifying which of two candidate stories most closely resembles an anchor story based on three dimensions: abstract themes, the sequence of events, and the final outcomes.

We developed three distinct but complementary methods to address this challenge. First, we fine-tuned a specialized story-embedding model using parameter-efficient techniques on synthetic data. Second, we utilized a "Distill-then-Embed" workflow, where a large language model extracts the essential narrative core of each story before computing similarity. Third, we employed direct zero-shot prompting to allow a high-reasoning model to compare the stories organically.

Our analysis reveals that each approach excels at different types of narrative comparisons, and their combination leads to robust performance. We demonstrate the importance of narrative distillation in removing surface-level distractors and the effectiveness of carefully engineered prompts in guiding language models to focus on narrative structure

1 Introduction

The Narrative Similarity Task (Hatzel et al., 2026) presents a fundamental challenge in natural language understanding: the ability to recognize identical story structures across superficially distinct texts. In this task, a system is presented with an "anchor" story and two candidates, and must determine which candidate shares a closer narrative affinity with the anchor. Unlike traditional semantic similarity, which often relies on lexical overlap, narrative similarity is defined by the alignment of three deep structural components:

- **Abstract Theme:** representing the defining constellation of problems and central ideas.
- **Course of Action:** which tracks the specific sequence of events, conflicts, and turning points.
- **Outcomes:** focusing on the final resolution and characters' fates rather than intermediate states.

A critical aspect of this task is the requirement to disentangle the "core story" from surface-level variations. Models must explicitly disregard distractors such as the concrete setting (e.g., a spaceship vs. a sailing ship), the specific names of characters and locations, and the writing style or level of detail. Success requires identifying that two stories are "the same" even if one describes a corporate takeover and the other a medieval siege, provided their thematic progression and resolutions align.

To address this complex abstraction problem, we propose and evaluate three distinct methodologies:

1. **Fine-tuning with QLoRA:** We adapt the story-emb model using parameter-efficient fine-tuning on synthetic data to learn robust representations of narrative structure.
2. **Distill-then-Embed:** We employ Gemini 1.5 Flash to strip away surface details and extract "narrative cores"—removing names and settings—before computing similarity via embeddings.
3. **Direct Prompting:** We leverage the reasoning capabilities of Gemini 2.5 Flash with engineered prompts to perform zero-shot comparisons of the narrative arcs.

2 Related Work

The study of narrative similarity is rooted in Structure-Mapping Theory (Gentner, 1983), which

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 3089–3094

posits that human analogy-making relies on aligning relational systems, such as causal chains, rather than matching independent attributes. Early computational efforts to operationalize this focused on unsupervised learning of event chains (Chambers and Jurafsky, 2008) and modeling latent character personas (Bamman et al., 2013) to capture the underlying roles within a story.

With the advent of deep learning, **Semantic Textual Similarity (STS)** research shifted toward dense vector representations. Standard Transformer-based models, such as SBERT (Reimers and Gurevych, 2019), have become the baseline for capturing semantic equivalence. However, these models are often optimized for short-range context and struggle with full narratives where similarity is defined by global structural coherence rather than local lexical overlap. To bridge this gap, long-document architectures like Longformer (Beltagy et al., 2020) have been introduced to aggregate information across dispersed narrative arcs.

Following the introduction of these long-context architectures, the landscape of computational narratology shifted significantly toward leveraging Large Language Models (LLMs) to parse and evaluate story structures. Recent studies have demonstrated that LLMs can effectively serve as proxies for human annotators in complex narrative tasks, such as narrative event segmentation (Michelmann et al., 2023) and the extraction of multi-layered narrative styles (Zhu et al., 2023). However, analogical reasoning across narratives—where surface details must be ignored in favor of abstract thematic alignment—remains a persistent challenge. To address this, recent frameworks have proposed using LLMs to decompose stories into abstract units prior to structural mapping, proving that explicit abstraction significantly enhances structural reasoning over raw end-to-end inference (Anonymous, 2026). Concurrently, the development of specialized representation models has advanced the field; notably, the introduction of *story-emb* demonstrated that models pre-trained specifically on fictional narratives capture holistic story elements far better than general-purpose sentence transformers (Hatzel and Biemann, 2024). Building on these theoretical and structural foundations, the SemEval-2026 Narrative Similarity task (Hatzel et al., 2026) challenges systems to distinguish between surface-level stylistic resemblance and deep, plot-based congruency. Our work builds directly upon this recent momentum, investigating whether LLM-driven distillation and specialized fine-tuning can systematically isolate the narrative core from surface-level distractors.

3 Methodology

Our approach to the Narrative Similarity task is designed to distinguish between surface-level coincidences, such as similar settings or character names, and deep structural alignment. To explore the most effective way to capture these alignments, we developed and evaluated three distinct strategies: *Narrative Distillation*, *Specialized Fine-Tuning*, and a *Zero-Shot Multi-Model Ensemble* (illustrated in Figure 1). Each method addresses narrative structure from a different perspective, ranging from explicit extraction of plot points to the use of high-reasoning models for organic comparison.

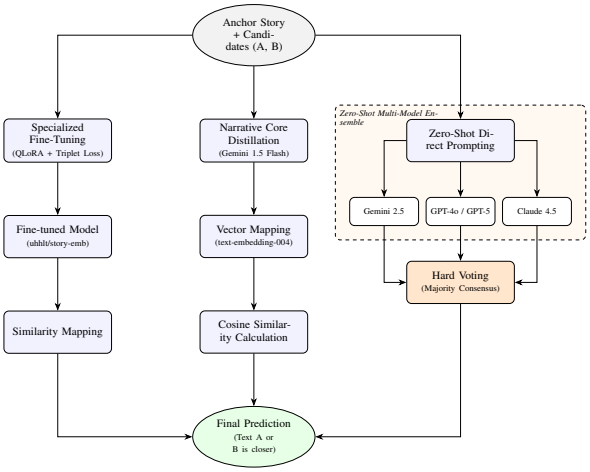

Figure 1: System Architecture: Workflow of the three proposed methods for narrative similarity detection, including the Distill-then-Embed pipeline, QLoRA fine-tuned embeddings, and the multi-model zero-shot ensemble.

3.1 Data and Triplet Construction

The model was fine-tuned using a dataset of narrative summaries sourced from the official SemEval-2026 Task 4 data release. To expand the training set and lower the barrier to entry, we incorporated the auxiliary synthetic dataset provided by the task organizers, which consists of 1,900 triples generated by LLMs. For the Triplet Margin Loss optimization, each training instance was explicitly structured as an anchor story, a structurally aligned positive candidate, and a dissimilar negative candidate. This configuration forces the embedding space to prioritize structural narrative congruency over mere lexical overlap.

3.2 Narrative Core Distillation (Distill-then-Embed)

To isolate the "Narrative Core" as defined in the task guidelines, we implement a distillation process. This stage acts as a structural filter to remove stylistic noise.

3.2.1 Feature Extraction

We utilize Gemini 1.5 Flash to process the raw story text and isolate its structural core. The model is provided with a targeted distillation prompt designed to strip away surface-level details (such as specific settings and character names) and extract an abstract summary focusing on three key dimensions:

- **Theme:** The underlying moral or conceptual conflict.
- **Course of Action:** The essential plot beats and turning points.
- **Outcome:** The final resolution and its implications.

The complete system prompt and extraction instructions used for this process are provided in Appendix A.B (see *Narrative Distillation Prompt*).

3.2.2 Vector Space Mapping

These distilled summaries are embedded into a high-dimensional vector space using the `text-embedding-004` model. We calculate the similarity between the Anchor (v_{anc}) and the candidate stories (v_a, v_b) using Cosine Similarity:

$$S(v_{anc}, v_i) = \frac{v_{anc} \cdot v_i}{\|v_{anc}\| \|v_i\|} \quad (1)$$

The candidate with the higher similarity score is designated as the preferred choice for this method.

3.3 Fine-Tuning via Triplet Loss (QLoRA)

To further refine our representations, we fine-tune the `uhhlt/story-emb` (Hatzel and Biemann, 2024) model—a model pre-trained specifically on narrative structures.

Optimization Strategy We use QLoRA (Quantized Low-Rank Adaptation), The complete configuration is provided in Appendix A.A (see *QLoRA Fine-Tuning Configuration*) for memory efficiency, allowing us to update the model's narrative understanding with minimal computational overhead.

Loss Function The model is trained using Triplet Loss, which optimizes the embedding space by ensuring that the distance between the Anchor (a) and a Positive example (p) is smaller than the distance between the Anchor and a Negative example (n) by a margin α:

$$\mathcal{L} = \max(d(a, p) - d(a, n) + \alpha, 0) \quad (2)$$

Computational Setup Fine-tuning was performed on a single NVIDIA A100 GPU via Google Colab. We trained for 5 epochs using the AdamW optimizer with a learning rate of $1e{-}5$, a batch size of 2, and a maximum sequence length of 1024 tokens. The Triplet Margin Loss was applied with $\alpha = 0.5$ and Euclidean distance ($p = 2$). The 4-bit NF4 quantization (with double quantization and `float16` compute dtype) combined with LoRA adapters ($r = 16$, $\alpha = 32$, dropout$= 0.05$) reduced the trainable parameter footprint substantially, enabling fine-tuning of the full model on a single accelerator.

3.4 Ensemble Strategy: Zero-Shot Multi-Model Voting

The core of our inference logic is a Zero-Shot Ensemble that aggregates the reasoning of multiple state-of-the-art Large Language Models (LLMs). This approach leverages the collective judgment of Gemini, GPT, and Claude.

3.4.1 Zero-Shot Prompting Logic

Each model in the ensemble is provided with the same *Zero-Shot prompt*. The prompt instructs the model to analyze the Anchor story alongside two candidates (Text A and Text B) and identify which candidate is narratively closer based on abstract plot structure. The full text of this prompt and the output specifications are provided in Appendix A.C (see *Direct Inference Prompt*). No examples are provided, ensuring the models rely on their internal understanding of narrative theory.

3.4.2 Hard Voting (Majority Consensus)

The ensemble employs a Hard Voting mechanism. Each model M_i produces a binary output $x_i \in \{0, 1\}$, where 1 indicates that Text A is closer and 0 indicates Text B is closer.

3.4.3 Resilience and Tie-Breaking

The system is designed to be fault-tolerant:

- **Missing Responses:** If a model fails to return a valid prediction due to API errors or safety filters, the majority is calculated based on the remaining active models.

- **Tie-Breaking:** In the event of a tie (e.g., in a 2-model valid response scenario), the system utilizes a predefined hierarchy, prioritizing the model that historically shows the highest alignment with the distilled narrative results.

3.5 Evaluation Metrics

In accordance with the SemEval-2026 Track A requirements, the primary evaluation metric is Accuracy, representing the ratio of correctly identified "closer" stories over the total number of instances in the test set. Although a k-fold cross-validation strategy could provide additional stability estimates, our evaluation follows the standard SemEval-2026 Task 4 protocol, which uses a fixed train/dev/test split released by the task organizers. All reported accuracies are measured on this official held-out test set.

4 Results

In this section, we present the performance of our narrative similarity detection models. We evaluate the accuracy of various approaches, ranging from zero-shot baselines and embedding-based methods to fine-tuned models and LLM ensembles. The task involves identifying which of two candidate stories is narratively closer to an anchor story based on abstract theme, course of action, and outcomes.

4.1 Baseline Performance

We adopted the official baseline provided by the task organizers (Hatzel et al., 2026), which utilizes a direct zero-shot prompting method with GPT-4o-mini. As shown in Table 1, this baseline yielded an accuracy of 0.55.[1] This result highlights that identifying narrative similarity—relying on structural rather than lexical cues—is a non-trivial challenge for models without specialized distillation or fine-tuning.

4.2 Embedding and Fine-Tuning Approaches

We explored two methods to improve representation learning for narratives:

- **Distill-Then-Embed:** This pipeline follows a sequential workflow: first, Gemini 1.5 Flash distills the narrative core by removing surface distractors, and then `text-embedding-004` is used for vectorization. Using cosine similarity as the metric, this approach achieved an accuracy of 0.66 (Hatzel et al., 2026).

- **QLoRA Fine-Tuning:** We fine-tuned the `uhhlt/story-emb` model using QLoRA (4-bit quantization) and Triplet Margin Loss to optimize the embedding space. This **strategy** outperformed the distillation-based pipeline, reaching an accuracy of 0.69. These results demonstrate the efficacy of training with a specialized loss function tailored for narrative structures (Reimers and Gurevych, 2019).

4.3 Large Language Model Evaluation

We further evaluated direct inference capabilities through a multi-model voting pipeline. In this framework, advanced LLMs act as expert annotators to determine narrative similarity. The Gemini 2.5 Flash model achieved a standalone accuracy of 0.71, surpassing both embedding-based techniques (Hatzel et al., 2026).

To maximize performance, we implemented multi-model ensembles. The results indicate that combining predictions from top-tier models yields the highest accuracy:

- An ensemble of Gemini 2.5, GPT-4o, and Claude Sonnet 4.5 achieved an accuracy of 0.74.

- The highest performance was observed with an ensemble of Gemini 2.5, GPT-5, and Claude Opus 4.5, all of which are state-of-the-art models released recently, reaching a peak accuracy of 0.75 .

Our findings suggest that while fine-tuning specialized embedding models (QLoRA) offers significant improvements over baselines, state-of-the-art LLMs and their ensembles currently provide the most accurate estimation of narrative similarity. Specifically, our best ensemble reached 75% accuracy, ranking 7th out of 47 participants in the competition.

4.4 Cost-Performance Trade-offs

While the zero-shot multi-model ensemble achieved the peak accuracy of 0.75, this approach

[1] We report the baseline accuracy as published by the task organizers; we did not re-run the baseline locally and therefore cannot rule out configuration-related variance (e.g., API token availability) on the reported value.

Method	Accuracy
Baseline (GPT-4o-mini, Zero-Shot)	0.55
Distill-Then-Embed (Gemini Embeddings)	0.66
Fine-Tuned QLoRA (story-emb)	0.69
Gemini 2.5 Flash (Direct Inference)	0.71
Ensemble (Gemini 2.5, GPT-4o, Claude Sonnet 4.5)	0.74
Ensemble (Gemini 2.5, GPT-5, Claude Opus 4.5)	**0.75**

Table 1: Comparative analysis of narrative similarity accuracy across different methods and ensembles.

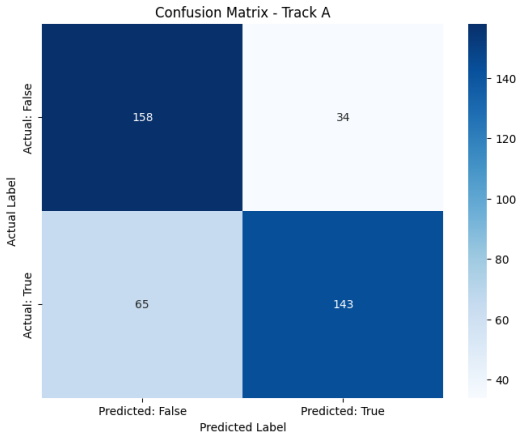

Figure 2: Confusion Matrix for Track A, detailing the actual vs. predicted labels for the best-performing ensemble configuration.

relies heavily on proprietary, state-of-the-art API endpoints (Gemini 2.5, GPT-5, Claude Opus 4.5). This introduces dependencies on external servers, increased inference latency, and higher computational costs per query. In contrast, our QLoRA fine-tuned *uhhlt/story-emb* model reached a competitive accuracy of 0.69 while operating entirely locally. This open-weight methodology eliminates API costs and significantly reduces latency, offering a more scalable and privacy-preserving alternative for large-scale narrative similarity tasks, despite a modest drop in zero-shot performance.

4.5 Error Analysis

To gain deeper insights into the performance of our approach, we analyzed the confusion matrix for Track A, generated from our highest-performing ensemble model (75% accuracy). The confusion matrix, presented in Figure 2, illustrates the distribution of the model's predictions across 400 evaluation pairs.

The matrix reveals that the model is highly effective at identifying the negative class, with only 34 False Positives compared to 158 True Negatives. However, there is a noticeable imbalance in the errors: the model produced nearly twice as many False Negatives (65) as False Positives (34). In the context of our binary classification task (determining if `text_a_is_closer`), this suggests that the ensemble is somewhat conservative and leans slightly toward predicting False when uncertain. Future work could involve adjusting the voting weights of the ensemble models to balance this threshold.

A Appendix: Implementation Details

For full reproducibility, our source code, dataset splits, Jupyter notebooks, and training scripts are publicly available on GitHub at: https://github.com/DVORA-AZARKOVICH/Narrative-Similarity.

A. QLoRA Fine-Tuning Configuration (Original Script)

```
bnb_config = BitsAndBytesConfig(
    load_in_4bit=True,
    bnb_4bit_use_double_quant=True,
    bnb_4bit_quant_type="nf4",
    bnb_4bit_compute_dtype=torch.float16
)

base_model = AutoModel.from_pretrained(
    model_name,
    quantization_config=bnb_config,
    device_map="auto"
)

base_model = prepare_model_for_kbit_training(base_model)

peft_config = LoraConfig(
    r=16,
    lora_alpha=32,
    target_modules=["q_proj", "v_proj", "k_proj", "o_proj",
    "gate_proj", "up_proj", "down_proj"],
    lora_dropout=0.05,
    bias="none",
    task_type=TaskType.FEATURE_EXTRACTION
)
```

B. Narrative Distillation Prompt

```
You are an expert annotator tasked with extracting
the 'Narrative Core' of a story.

### DEFINITIONS
Extract the core based ONLY on:
1. **Abstract Theme**: The defining constellation of problems and
central ideas.
2. **Course of Action**: The sequence of events, actions, conflicts,
and turning points.
3. **Outcomes**: The results of the plot (resolution, fates).

### WHAT TO IGNORE
Explicitly REMOVE and IGNORE:
* The concrete setting (e.g., Sci-Fi, Western, dates, locations).
* Names of characters.
* The style of writing.

### OUTPUT INSTRUCTION
Rewrite the story summary to include ONLY the Abstract Theme, Course of
Action, and Outcomes. Do not analyze, just describe the narrative core
neutrally.
```

C. Direct Inference Prompt

```
You are an expert annotator tasked with identifying narrative
similarity between stories.
Your goal is to determine which of two candidate stories
(Text A or Text B) is narratively closer to an Anchor story.

### DEFINITIONS OF NARRATIVE SIMILARITY
You must evaluate similarity based ONLY
on the following three core aspects:
1. **Abstract Theme**: The defining constellation of problems,
central ideas, and core motifs.
2. **Course of Action**: The sequence of events, actions, conflicts,
turning points.
3. **Outcomes**: The results of the plot at the end of the text.

### INSTRUCTIONS
1. Analyze the Anchor.
2. Compare Text A to Anchor.
3. Compare Text B to Anchor.
4. Decide which text is narratively
closer overall.

### INPUT DATA
**Anchor Text:** {{INSERT_ANCHOR_TEXT_HERE}}
**Text A:** {{INSERT_TEXT_A_HERE}}
**Text B:** {{INSERT_TEXT_B_HERE}}

### OUTPUT FORMAT
Provide your response in valid JSON format:
{
  "reasoning": "Brief explanation.",
  "text_a_is_closer": boolean
}
```

References

Anonymous. 2026. Enhancing structural mapping with llm-derived abstractions for analogical reasoning in narratives. In *Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing*.

David Bamman, Brendan O'Connor, and Noah A. Smith. 2013. Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 352–361, Sofia, Bulgaria. Association for Computational Linguistics.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

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.

Dedre Gentner. 1983. Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7(2):155–170.

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. Association for Computational Linguistics.

Sebastian Michelmann and 1 others. 2023. Llms as approximations of human annotators in narrative event segmentation. *arXiv preprint*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Yuchen Zhu and 1 others. 2023. Exploring complex narrative analysis tasks with large language models. *arXiv preprint*.