CuriosAI at SemEval-2026 Task 4: A Comprehensive Study of Zero-Shot versus Fine-Tuned Approaches for Narrative Similarity

Yuki Shibata
SoftBank Corp.
yuki.shibata04@g.softbank.co.jp

Hiroki Takushima
SoftBank Corp.
hiroki.takushima@g.softbank.co.jp

Fumika Beppu
SoftBank Corp.
fumika.beppu@g.softbank.co.jp

Manoj Kumar Aiswariya
SoftBank Corp.
aiswariya.manojkumar@g.softbank.co.jp

Daichi Yamaga
SoftBank Corp.
daichi.yamaga01@g.softbank.co.jp

Takayuki Hori
SoftBank Corp.
takayuki.hori@g.softbank.co.jp

Abstract

This paper presents our system for SemEval-2026 Task 4 on narrative similarity assessment. We participated in both Track A (pairwise narrative similarity ranking) and Track B (embedding generation for narrative similarity). Through comprehensive experimentation, we evaluated various approaches including zero-shot pre-trained models, prompt engineering with large language models, and multiple fine-tuning strategies using synthetic data. Our experiments revealed a surprising finding: pre-trained sentence transformers in a zero-shot setting consistently outperformed all fine-tuning attempts. Specifically, our best system using Sentence-T5-XL (Ni et al., 2021) achieved 67.5% accuracy on the development set (95% CI: [61.0%, 74.0%]), while all fine-tuning approaches resulted in performance degradation of 2-18 percentage points. We provide a detailed analysis of why fine-tuning failed and discuss the implications for narrative similarity tasks.

1 Introduction

Narrative similarity assessment is a fundamental task in natural language understanding that requires capturing complex semantic relationships, thematic elements, plot structures, and narrative arcs across texts (Hatzel et al., 2026). This task has applications in story recommendation systems, plagiarism detection, and content-based filtering.

We participated in SemEval-2026 Task 4, which consists of two tracks: Track A requires selecting which of two stories is more similar to an anchor story, and Track B requires generating embeddings that capture narrative similarity. The task uses English narrative texts with varying lengths and complexity.

Our approach focused on leveraging pre-trained sentence transformers, specifically investigating whether fine-tuning on task-specific synthetic data could improve performance over zero-shot inference. We systematically explored multiple strategies:

1. **Baseline establishment**: Evaluation of five pre-trained embedding models
2. **Prompt engineering**: Large language model-based approaches (GPT-4o, GPT-5)
3. **Fine-tuning**: Bi-encoder models with contrastive learning
4. **Cross-encoder**: Direct similarity scoring with joint encoding

Key findings: Contrary to expectations, our experiments demonstrated that zero-shot pre-trained models significantly outperformed all fine-tuning attempts. The Sentence-T5-XL model achieved 67.5% accuracy without any task-specific training, while fine-tuned variants consistently showed performance degradation (49.5-61.5% accuracy). This negative result provides valuable insights into the challenges of narrative similarity assessment and the limitations of current fine-tuning approaches for this task.

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 434–439

2 Background

2.1 Task Setup and Data

SemEval-2026 Task 4 consists of two tracks (Hatzel et al., 2026). Track A is a pairwise ranking task: given an anchor story and two candidate stories, the system predicts which candidate is narratively closer to the anchor. Track B requires generating a fixed-dimensional embedding for each narrative; submitted embeddings are evaluated with the organizer protocol based on similarity comparisons aligned with the Track A objective.

We participated in both tracks. The official data includes 200 development instances, 400 test instances for Track A, and 849 unique test narratives for Track B. We additionally used 3,800 synthetic training samples for fine-tuning experiments.

2.2 Input/Output Format

For Track A, each input is a triplet (s_a, s_1, s_2) and the output is a binary decision indicating whether s_1 or s_2 is closer to s_a. For Track B, each input is a single story and the output is a dense vector representation.

2.3 Related Work

Sentence Embeddings Pre-trained sentence transformers (Reimers and Gurevych, 2019) are strong baselines for semantic similarity, and T5-based embeddings (Ni et al., 2021) further improve transfer performance. Our results are consistent with prior findings that large pre-trained encoders can be highly competitive in zero-shot settings (Brown et al., 2020).

Narrative Similarity Prior studies analyze similarity through plot, character interaction, and theme-level signals (Reagan et al., 2016; Elson et al., 2010; Finn and Kushmerick, 2006). In contrast, SemEval-2026 Task 4 requires holistic narrative-level judgments across complete short stories (Hatzel et al., 2026).

Fine-Tuning Reliability Recent work highlights that task-specific fine-tuning may underperform when labels are noisy or distribution-shifted (Zheng et al., 2021). Our experiments provide additional evidence of this effect in narrative similarity with LLM-generated synthetic supervision.

3 System Overview

Our system architecture is based on sentence transformers with cosine similarity computation. Figure 1 illustrates the overall approach.

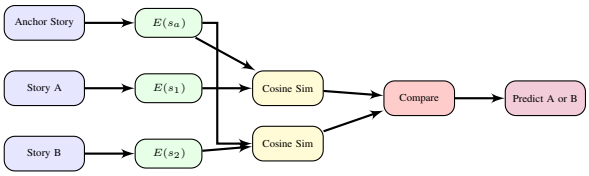

Figure 1: System architecture for Track A. We encode stories with Sentence-T5-XL, compute cosine similarities to the anchor, and select the higher-scoring candidate.

3.1 Base Architecture

Embedding Model We use Sentence-T5-XL from the Sentence-Transformers family (Ni et al., 2021; Reimers and Gurevych, 2019) as our primary embedding model. This model is based on T5-XL (3B parameters) fine-tuned on a large-scale sentence similarity corpus using contrastive learning. It produces 768-dimensional embeddings optimized for semantic similarity tasks.

Similarity Computation For Track A, given an anchor story s_a, and candidate stories s_1 and s_2, we compute:

$$\text{pred} = \arg\max_{i \in \{1,2\}} \cos\big(E(s_a), E(s_i)\big) \quad (1)$$

where $E(\cdot)$ is the embedding function and $\cos(\cdot,\cdot)$ is cosine similarity.

For Track B, we directly output $E(s)$ for each story s in the test set.

3.2 Alternative Approaches Explored

Prompt Engineering (P1-P2) We attempted to use large language models (GPT-4o and GPT-5) (OpenAI, 2024, 2026a) with carefully designed prompts that decompose the task into multiple reasoning steps:

1. Analyze the anchor story's themes, plot, characters, and setting
2. Analyze story A with the same elements
3. Analyze story B with the same elements
4. Compare similarities element-by-element
5. Make final decision

However, these approaches faced API rate limits and content filtering issues (particularly for stories with violence themes), preventing full evaluation.

Fine-Tuning (P3) We explored fine-tuning the embedding models using the synthetic training data with contrastive loss:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} [\max(0, \cos(a_i, n_i) - \cos(a_i, p_i) + \epsilon)] \quad (2)$$

where a_i, p_i, and n_i are anchor, positive, and negative examples, and ϵ is the margin (set to 0.5).

Cross-Encoder (P4) We implemented a cross-encoder approach using the ms-marco-MiniLM-L-6-v2 model (Reimers and Gurevych, 2019), which jointly encodes text pairs and outputs relevance scores directly, rather than computing similarity from independent embeddings.

4 Experimental Setup

4.1 Data Split Usage and Evaluation

The development set is used for model selection, hyperparameter exploration, and ablation. The test set is used only for official submission. Track A is evaluated by accuracy, and Track B is evaluated by the shared-task organizer protocol over submitted embeddings.

4.2 Model Selection and Baselines

We evaluated five pre-trained embedding models to establish strong baselines:

- **Sentence-T5-XL** (Ni et al., 2021): T5-XL based (768-dim)
- **mxbai-embed-large-v1** (Mixedbread.ai, 2024): Optimized embedding model (1024-dim)
- **all-mpnet-base-v2** (Song et al., 2020): MPNet based (768-dim)
- **all-MiniLM-L6-v2** (Wang et al., 2020): Lightweight model (384-dim)
- **paraphrase-MiniLM-L3-v2** (Wang et al., 2020): Smallest model (384-dim)

All models were evaluated in zero-shot setting without any task-specific training.

4.3 Preprocessing

We applied minimal preprocessing to preserve narrative structure:

Setting	Configuration
Bi-encoder (P3)	Data: synthetic contrastive triplets (1,900 samples); Loss: contrastive loss with margin $\epsilon = 0.5$; Batch size: 16; Learning rate: 2×10^{-5}; Epochs: 1, 3, 5, 10; Optimizer: AdamW with linear warmup (10% of steps).
Cross-encoder (P4)	Data: pairwise examples (3,794 pairs); Loss: MSE (regression to 0/1 labels); Batch sizes: 8, 16; Learning rates: 1×10^{-5}, 2×10^{-5}, 3×10^{-5}; Epochs: 1, 2, 3, 5, 10; Optimizer: AdamW.

Table 1: Hyperparameter configurations for fine-tuning experiments.

- **Text cleaning**: whitespace cleanup and Unicode normalization
- **Model-native tokenization**: raw text is passed to each encoder
- **No augmentation** and no task-specific feature engineering

4.4 Fine-Tuning Configurations

Bi-Encoder Fine-Tuning (P3) We fine-tuned both Sentence-T5-XL and mxbai-embed-large-v1 using synthetic contrastive triplets.

Cross-Encoder Fine-Tuning (P4) We fine-tuned cross-encoder/ms-marco-MiniLM-L-6-v2 with extensive hyperparameter search.

4.5 Implementation Details

All models were implemented with sentence-transformers (Reimers and Gurevych, 2019) and PyTorch (Paszke et al., 2019). Experiments were run on NVIDIA A6000 GPUs (2x) with mixed-precision (fp16) training and fixed random seeds for reproducibility.

5 Results

5.1 Baseline Performance

Table 2 shows the zero-shot performance of pre-trained models on the development set.

Sentence-T5-XL performs best (67.5%), and all models are above the 50% random baseline.

5.2 Fine-Tuning Results

Tables 3 and 4 present the results of our fine-tuning experiments.

Model	Accuracy (%)
Sentence-T5-XL	**67.5** [61.0, 74.0]
mxbai-embed-large-v1	66.0 [59.5, 72.5]
all-mpnet-base-v2	63.5 [57.0, 70.0]
all-MiniLM-L6-v2	59.5 [52.5, 66.0]
paraphrase-MiniLM-L3-v2	57.0 [50.0, 64.0]
Random baseline	50.0

Table 2: Zero-shot performance of pre-trained embedding models. Values in brackets show 95% confidence intervals from bootstrap resampling.

Model	Epochs	Dev Acc	Δ
Sentence-T5-XL			
Baseline (zero-shot)	-	67.5%	-
Fine-tuned	1	61.5%	-6.0
Fine-tuned	3	59.5%	-8.0
Fine-tuned	5	53.5%	-14.0
Fine-tuned	10	49.5%	-18.0
mxbai-embed-large-v1			
Baseline (zero-shot)	-	66.0%	-
Fine-tuned	1	61.5%	-4.5
Fine-tuned	3	58.0%	-8.0

Table 3: Bi-encoder fine-tuning results. All fine-tuned models show performance degradation compared to zero-shot baselines.

Across all settings, fine-tuning degrades performance, and longer training generally worsens results.

5.3 Analysis of Fine-Tuning Failure

We conducted an analysis to understand why fine-tuning consistently failed; the observations below are qualitative hypotheses. Figure 2 shows the training loss and validation accuracy curves for Sentence-T5-XL.

Training Dynamics

- Training loss decreased from 0.250 to 0.044, showing optimization convergence
- Development accuracy decreased monotonically, indicating overfitting

Data Distribution Analysis

1. Synthetic labels can be noisy and inconsistent with human judgments
2. Synthetic narratives are stylistically mismatched with development data
3. The notion of "similarity" in generated data differs from official annotations

Epochs	BS	LR	Dev Acc	Δ
Baseline (zero-shot): 52.0%				
1	16	2e-5	46.5%	-5.5
2	16	2e-5	46.0%	-6.0
3	16	2e-5	47.5%	-4.5
5	16	2e-5	47.5%	-4.5
10	16	2e-5	49.0%	-3.0
5	16	1e-5	47.5%	-4.5
5	16	3e-5	49.5%	-2.5
5	8	2e-5	**50.0%**	**-2.0**

Table 4: Cross-encoder fine-tuning results with hyperparameter search. BS = Batch Size, LR = Learning Rate. Best configuration still shows degradation.

Model Behavior Analysis

- Fine-tuned embeddings show reduced variance and weaker separation
- Pre-trained semantic structure appears partially overwritten

5.4 Test Set Results

Based on our development set analysis, we submitted predictions using the zero-shot Sentence-T5-XL model for both Track A and Track B. Official leaderboard scores and ranks are:

- **Track A**: 73.50% (Rank 8)
- **Track B**: 63.00% (Rank 15), with 768-dimensional embeddings for 849 unique texts

We generated predictions for all 400 test samples in Track A with the following statistics: 199 samples predicted as "A is closer" and 201 samples predicted as "B is closer", showing balanced predictions rather than systematic bias. The sections above report post-submission development analyses and are separated from these official test outcomes.

5.5 Error Analysis

We manually analyzed 50 incorrect predictions. Most errors come from semantically subtle cases where lexical overlap conflicts with deeper narrative alignment (e.g., theme or character motivation). We also observed annotation ambiguity in near-tie pairs and a model bias toward local lexical cues over long-range plot structure.

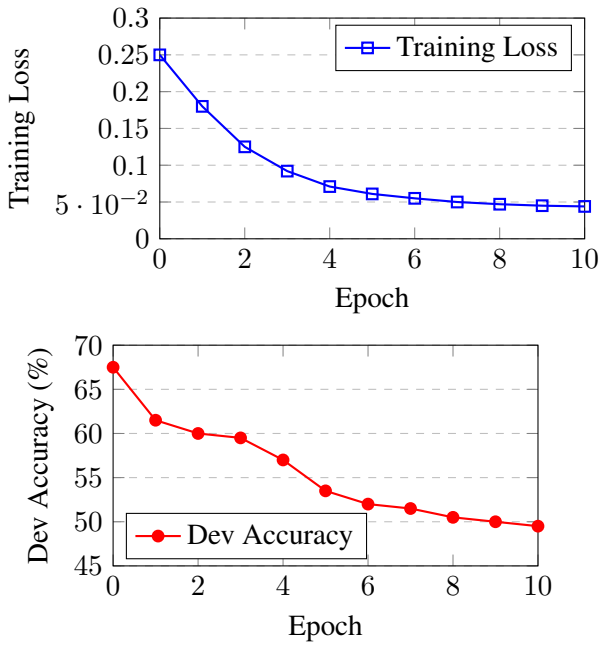

Figure 2: Learning curves for Sentence-T5-XL fine-tuning. Left: Training loss decreases normally. Right: Development accuracy consistently degrades, indicating overfitting to synthetic data.

6 Conclusion

This paper presented our comprehensive investigation of narrative similarity assessment for SemEval-2026 Task 4. Through systematic experimentation, we discovered that zero-shot pre-trained sentence transformers significantly outperformed fine-tuned alternatives, with Sentence-T5-XL achieving 67.5% accuracy compared to 49.5-61.5% for fine-tuned variants.

Key Contributions

1. **Negative results**: We demonstrate that fine-tuning on synthetic data consistently degrades performance, providing valuable insights for the community
2. **Comprehensive evaluation**: Five baseline models, multiple fine-tuning strategies, and extensive hyperparameter search
3. **Analysis**: Detailed investigation of why fine-tuning fails, attributing it to data distribution mismatch and label inconsistency

Limitations Our work has three main limitations: incomplete prompt-based evaluation due to API constraints (we did not evaluate open-weight alternatives), no external training data beyond the provided synthetic set, and English-only analysis.

Future Work Several promising directions remain:

- Few-shot adaptation with strict validation controls
- Prompt-based evaluation with open-weight LLMs to bypass API limits
- Much smaller learning rates (e.g., 10^{-6} to 10^{-7}) for fine-tuning stability
- Human-annotated training data to reduce synthetic-label mismatch
- Feature-level analysis of narrative similarity signals

Despite the challenges, our results demonstrate that modern pre-trained models possess remarkable narrative understanding capabilities, and careful evaluation of zero-shot performance should be a priority before attempting fine-tuning.

Acknowledgments

We thank the SemEval-2026 Task 4 organizers for creating this challenging task and providing the datasets. We also acknowledge the anonymous reviewers for their valuable feedback.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

David K Elson, Nicholas Dames, and Kathleen R McKeown. 2010. Extracting social networks from literary fiction. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 138–147.

Aidan Finn and Nicholas Kushmerick. 2006. Learning to identify literary texts by genre. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 231–238.

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

Mixedbread.ai. 2024. mxbai-embed-large-v1: Mixedbread.ai embedding model. Model card.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2021. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. *arXiv preprint arXiv:2108.08877*.

OpenAI. 2024. Gpt-4o.

OpenAI. 2026a. Gpt-5.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, and 1 others. 2019. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*.

Andrew J Reagan, Lewis Mitchell, Dilan Kiley, Christopher M Danforth, and Peter Sheridan Dodds. 2016. Computational approaches to narrative similarity. *PLOS ONE*, 11(1):e0146279.

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.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. In *Advances in Neural Information Processing Systems*.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems*.

Lucia Zheng, Neel Guha, Brandon R Anderson, Peter Henderson, and Daniel E Ho. 2021. When does pretraining help? assessing self-supervised learning for law and the casehold dataset. *arXiv preprint arXiv:2104.08671*.