CophiWue at SemEval-2026 Task 4: Symbolic Narrative Profiling with Taxonomy-Guided Extraction and Contrastive Fine-Tuning

Leonard Konle[1], **Fotis Jannidis**[1],
[1]Julius-Maximilians-Universität Würzburg
Correspondence: leonard.konle@uni-wuerzburg.de

Abstract

We present our system for SemEval-2026 Task 4, focusing primarily on Track B (narrative embedding). Our approach, the **Decompose & Align Cycle**, converts each story into a structured *NarrativeProfile* consisting of abstract themes, a five-step course of action, and an outcome. We then build a *NarrativeTaxonomy* from these initial extractions via agglomerative clustering, and use the resulting controlled vocabularies to guide a second extraction pass, producing terminologically standardized profiles across the full dataset. Finally, we contrastively fine-tune the Qwen3-Embedding-8B model on profile text representations using TripletLoss, deriving story embeddings from this fine-tuned model. For Track A, we adapt the task's provided baseline script by substituting Gemini 3 Pro as the judge, using the organizers default prompt on raw story texts.

1 Introduction

SemEval-2026 Task 4 (Hatzel et al., 2026) formalizes the task of detecting narrative similarity with two complementary tracks. **Track A** asks which of two candidate stories is more narratively similar to an anchor (binary preference). **Track B** requires producing dense vector representations of stories such that narratively similar stories cluster together in embedding space.

The task organizers explicitly define narrative similarity along three dimensions: *abstract theme* (ideas and motifs), *course of action* (the event sequence), and *outcomes* (the resolution). Our system better reflects these dimensions by extracting them explicitly rather than relying on a model to discover them implicitly from raw text.

Track B is our primary focus. Our hypothesis is that an embedding trained on abstract *profile representations* rather than raw text will better generalize across the genre, vocabulary, and stylistic variation present in a Wikipedia story corpus.

For Track A, we opted for a simpler approach: we adapted the task organizers' baseline script by replacing the default LLM with Gemini 3 Pro, using the script's default prompt unchanged on the raw story texts.

On the official test set, our Track A system (Gemini 3 Pro zero-shot) achieves **69.50%** accuracy (rank 16 of 44 teams), above the GPT-4o-mini task baseline of 67%. Our Track B system achieves **61.50%** (rank 18 of 27 teams), above the sentence-embedding baseline (58.50%) but below the narrative-specific story-emb baseline (63.25%). Taxonomy construction reveals a 58:1 compression ratio for outcomes, confirming that narrative resolutions cluster into a small set of universal archetypes. Code is available at `https://github.com/LeKonArD/konle-narrative-similarity`.

Contributions.

1. A structured *NarrativeProfile* representation capturing theme, course of action, and outcome separately via constrained LLM extraction.

2. An iterative *Decompose & Align Cycle*: extract free-form profiles, cluster them into a *NarrativeTaxonomy*, and re-extract using the taxonomy as a controlled vocabulary, producing consistent terminology across the dataset.

3. Contrastive fine-tuning of a large embedding model on profile text representations using TripletLoss and QLoRA.

2 Task Description

Data. The dataset consists of Wikipedia story summaries (films, novels, etc.). The development set contains 200 labeled triples for Track A and 479 texts for Track B. The test set contains 400 triples for Track A and 849 texts for Track B.

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 3479–3484

Track A. Each example is a triple (anchor, text_a, text_b) with a binary label indicating whether text_a or text_b is more narratively similar to the anchor. Similarity is defined along three dimensions: abstract theme, course of action, and outcomes. Performance is measured by accuracy.

Track B. Each example is a single story text. Systems must produce a fixed-size embedding per story. Performance is evaluated by how well the cosine similarity ordering of embeddings aligns with the human-judged similarity ordering from Track A.

3 System Description

3.1 Track A: LLM Judgment

Track A was not our focus. We adapted the task organizers' baseline script (`track_a.py`) by substituting **Gemini 3 Pro** (`google/gemini-3-pro-preview`) as the LLM backend. The **default prompt from the organizers' script** was used unchanged: the model receives the three raw story texts (anchor, text A, text B) and returns a binary choice (A or B). No story decomposition or training is involved.

3.2 Track B: Decompose & Align Cycle

Track B is our main contribution. Raw-text embeddings conflate surface style, genre vocabulary, and narrative structure, making it hard for a model to distinguish stories that share a plot arc but differ in setting. By explicitly extracting theme, course of action, and outcome—the three dimensions along which the task defines narrative similarity—we train and embed on the signal that directly reflects the evaluation criterion, rather than leaving the model to discover it from raw text. The pipeline has five steps (Table 1).

	Component	Input	Output
1	MinerAgent (Gemini Flash)	Raw text	Initial profiles
2	CuratorAgent (Qwen3-Emb)	Profiles	NarrativeTaxonomy
3	MinerAgent (guided)	Text + taxonomy	Standardized profiles
4	TripletLoss / QLoRA	Profiles + labels	Fine-tuned model
5	Embed	Profiles + model	Track B embeddings

Table 1: Track B pipeline. Standardized profiles (Step 3) feed both Step 4 (fine-tuning) and Step 5 (embedding).

3.2.1 Step 1: Profile Extraction

The **MinerAgent** (`agents/miner.py`) converts each raw story text into a structured **NarrativeProfile** using *Gemini 3 Flash* via OpenRouter at temperature $= 0.0$:

ABSTRACT_THEME 2–5 abstract keywords capturing universal concepts and motifs (e.g., *Revenge, Betrayal, Forbidden love*). Concrete settings such as "WWII" or "space adventure" are explicitly excluded by the prompt.

COURSE_OF_ACTION Exactly five chronological key events describing the narrative arc, formulated in present tense with specific names and locations abstracted away.

OUTCOME A single sentence describing how the central conflict resolves.

Extraction is constrained by a JSON schema enforced via the API's structured output mode (`abstract_theme`: 2–5 strings; `course_of_action`: 5 strings; `outcome`: 1 string), eliminating post-processing failures. Each profile is serialized as:

`Theme:` $[t_1, t_2, \ldots]$.
`Actions:` $[a_1] \rightarrow \cdots \rightarrow [a_5]$.
`Outcome:` $[o]$.

Worked Example. Consider the following story (from the development set):

> Albert Loriflan, a waiter in a Paris cafe, unexpectedly inherits a large sum of money from a wealthy relative. His unscrupulous boss, Philibert, refuses to release him from his long-term contract in the hope that Albert will buy him off with a large payment. But Albert refuses, and continues to work at the cafe even though he is now very rich. Before long he falls in love with Philibert's daughter Yvonne.

Step 1 (free-form extraction) produces:

THEMES *Greed, Power struggle, Social mobility, Forbidden love*

COURSE OF ACTION
1. A low-status worker unexpectedly gains immense wealth.
2. An authority figure attempts to exploit the worker's new fortune through legal entrapment.
3. The worker refuses to yield to financial extortion and maintains his humble position.
4. The worker develops a romantic interest in a relative of his antagonist.
5. The power dynamic shifts as the worker uses his persistence to challenge the antagonist's control.

OUTCOME *The protagonist maintains his integrity by refusing to be exploited, ultimately complicating the antagonist's leverage through an unexpected romantic connection.*

3.2.2 Step 2: Taxonomy Construction

A key problem with open-ended LLM extraction is *terminological drift*: the same narrative concept appears under different surface forms across stories (e.g., "betrayal by a friend", "trusted ally turns traitor", "friend's treachery"). Without standardization, stories sharing the same underlying theme can appear unrelated in embedding space.

The **CuratorAgent** (`agents/curator.py`) addresses this by building a **NarrativeTaxonomy** – three controlled vocabularies, one per profile component – via **agglomerative clustering** of the extracted terms.

Algorithm. For each component (themes, actions, outcomes), we collect all unique terms from Step 1, embed them with Qwen3-Embedding-8B, and run agglomerative clustering (average linkage, cosine metric) with thresholds $\theta \leq 0.35$ / 0.50 / 0.55 respectively (tighter for themes, looser for outcomes which admit fewer archetypes). The shortest term in each cluster becomes the canonical representative in `UPPERCASE` form.

The result is a `NarrativeTaxonomy` mapping every raw term to its canonical label for all three components. Typical vocabulary reduction on the development set is shown in Table 2.

Component	Raw	Canonical	Ratio
Themes	2,822	230	12:1
Actions	8,676	333	26:1
Outcomes	1,736	30	58:1

Table 2: Vocabulary reduction through agglomerative clustering (Step 2), built from 1,738 stories (combined Track A and Track B).

The compression ratios vary systematically by component type. Outcomes compress most aggressively (57.9:1), reflecting the universal character of narrative resolutions: a small set of archetypes – e.g., `CYCLE_OF_VIOLENCE_ENDS_IN_FATALITY_RESULTING_IN`, `IMMEDIATE_THREAT_IS_NEUTRALIZED_THROUGH_EXTREME_PHYSICAL_FORCE`, `JOURNEY_CONCLUDES_WITH_PARTICIPANTS_GAINING_NUANCED_UNDERSTANDING_OF` – accounts for the vast majority of endings. Actions compress moderately (26:1), as the same plot event can be phrased in many syntactically distinct ways. Themes compress least (12:1), reflecting genuine diversity in the thematic landscape of the Wikipedia story corpus.

3.2.3 Step 3: Taxonomy-Guided Re-Extraction

With the taxonomy in hand, we re-run extraction with all three canonical vocabularies injected into the prompt:

```
THEME VOCABULARY (use these
when applicable): REVENGE,
REDEMPTION, BETRAYAL, ... Only
introduce NEW terms if none
fits.
```

This *soft constraint* lets the LLM prefer canonical terms while retaining flexibility for genuinely novel narratives, producing *standardized profiles* with consistent terminology across the dataset.

Standardized Example. Continuing the worked example from Section 3.2.1, Step 3 maps the free-form profile to:

THEMES `CLASS_TRANSITION`, `CONTRACTUAL_LOOPHOLE`, `BUREAUCRATIC_IRONY`, `AMBITION`, `FORBIDDEN_LOVE`

COURSE OF ACTION
1. `A_LOW-STATUS_WORKER_UNEXPECTEDLY_GAINS_IMMENSE_WEALTH`
2. `BUREAUCRATS_EXISTING_COMMITMENT_TO_SOCIALLY_ADVANTAGEOUS_ENGAGEMENT_CREATES`
3. `ACQUISITION_TRIGGERS_RESENTMENT_AND_HOSTILITY_FROM_SOCIAL_SUPERIOR`
4. *"Protagonist maintains original employment despite new financial status"* (free-form)
5. *"Protagonist develops romantic feelings for the antagonist's daughter"* (free-form)

OUTCOME `BOND_OF_TRUST_AND_MUTUAL_RESPECT_DEVELOPS_BETWEEN`

Actions 1 and 3 illustrate the primary benefit of standardization: *Social mobility* is mapped to `CLASS_TRANSITION`, `FORBIDDEN_LOVE` got a canonical status and `CONTRACTUAL_LOOPHOLE` emerged from a different story, but got attached to this one. Labels shared with hundreds of other stories in the corpus, ensuring that narratives with a similar plot converge to the same representation regardless of how the LLM originally phrased them. Actions 4–5, having no close

canonical match, are retained as free-form text, illustrating the soft-constraint flexibility described above. Not every label is a clean fit: Action 2 and the outcome receive high-frequency generic labels, a failure mode discussed in Section 6.

A further limitation is visible here: the Wikipedia summary is incomplete and omits the film's resolution (Berger, 1931). In *The Little Cafe* (1931), Albert ultimately marries Yvonne, resolving the central conflict between him and Philibert — an outcome that would have grounded Actions 4–5 in a canonical label rather than free-form text, and produced a more informative outcome representation. Truncated summaries are a recurring source of incomplete profiles in our dataset.

Together, Steps 1 through 5 form the **Decompose & Align Cycle**: *Extract* → *Build Taxonomy* → *Re-Extract with Vocabulary*. In principle, the cycle can be iterated; we find one iteration sufficient for the development set.

3.2.4 Step 4: Contrastive Fine-Tuning

We fine-tune **Qwen/Qwen3-Embedding-8B** (Zhang et al., 2025) on narrative similarity triplets derived from the Track A training data.

Triplet Construction. For each labeled Track A example (anchor, text_a, text_b, label), we look up the standardized profiles from Step 3 and form a triplet, where winner/loser are determined by the binary label. Crucially, the model is trained on *profile representations*, not raw story texts, so it learns to distinguish narrative similarity at the abstract structural level.

Training. We use TripletLoss with cosine distance and margin $= 0.5$:

$$\mathcal{L} = \max(0,\ d(a,p) - d(a,n) + 0.5) \quad (1)$$

where d is cosine distance, a is the anchor story's profile, p is the profile of the story labeled as more similar (the Track A winner), and n is the profile of the other candidate (the Track A loser). Training uses the sentence-transformers library. The first 30 examples are held out as a validation set; the model checkpoint with highest validation triplet accuracy is retained.

We use **QLoRA** (Dettmers et al., 2023): 4-bit NF4 quantization with LoRA adapters ($r = 16$, $\alpha = 32$, dropout 0.1) applied to all attention and MLP projection matrices (q, k, v, o, gate, up, down projections), enabling fine-tuning of the 8B model on a single NVIDIA RTX A6000 (48 GB). The QLoRA implementation uses PEFT and bitsandbytes. Hyperparameters: 3 epochs, batch size 16, learning rate 3×10^{-4}, 100 warmup steps with linear decay.

3.2.5 Step 5: Embedding

Each story's standardized profile text is encoded by the fine-tuned model. For the QLoRA variant, mean pooling over masked last hidden states is applied followed by L2-normalization, yielding a single fixed-size vector per story comparable via cosine similarity.

4 Experimental Setup

LLM Configuration. Track B profile extraction (Steps 1 and 3) uses *Google Gemini 3 Flash* (`google/gemini-3-flash-preview`) via the OpenRouter API at temperature $= 0.0$, with JSON schema enforcement. Gemini 3 Flash was chosen for extraction to reduce API cost across the large number of per-story calls required. Track A uses *Google Gemini 3 Pro* (`google/gemini-3-pro-preview`) via the OpenRouter API with the organizers' default prompt and structured JSON output. Gemini 3 Pro was selected for Track A as the most capable model available on OpenRouter at submission time, where a single call per triple makes the higher cost acceptable.

Embedding and Clustering Model. Steps 2, 4, and 5 all use **Qwen/Qwen3-Embedding-8B**.

Taxonomy Thresholds. Clustering thresholds θ of 0.35 / 0.50 / 0.55 for themes, actions, and outcomes were set by manual inspection of cluster quality, checking that semantically distinct concepts (e.g., `HUBRIS` vs. `HEROISM`) remained in separate clusters while surface paraphrases merged.

Submitted Variants. Track A: Gemini 3 Pro zero-shot (default organizers' prompt). Track B: taxonomy-guided profiles + QLoRA fine-tuned Qwen3-Embedding-8B (NF4 4-bit, $r = 16$, $\alpha = 32$).

Evaluation Measures. Track A performance is measured by **accuracy**: the fraction of triples $(anchor, A, B)$ for which the system selects the correct candidate. Track B systems submit one fixed-size embedding per story; performance is evaluated by computing cosine similarity between

all pairs and measuring how well the resulting similarity ordering agrees with the human-judged preference labels from Track A—effectively testing whether the embedding geometry respects the narrative similarity structure annotated in the data.

5 Results

System	Track A	Track B
CophiWue (ours)	**69.50**	**61.50**
Best system	78.00	72.00
GPT-4o-mini / story-emb (task)	67.00	63.25
Random baseline	50.00	50.00

Table 3: Official test set accuracy (%). Best system: COGNAC (both tracks). Task LLM/embedding baselines from Hatzel et al. (2026).

Our Track A system ranks 16th out of 44 teams (69.50%), exceeding the GPT-4o-mini task baseline (67.00%) by 2.5 points. Our Track B system ranks 18th out of 27 teams (61.50%), above the generic sentence-embedding baseline (all-MiniLM-L6-v2, 58.50%) but below the narrative-specific story-emb baseline (63.25%). The gap to the Track B leader (72.00%) is substantial.

6 Analysis

Profile Representations and Terminological Consistency. Training on narrative profiles rather than raw story texts avoids overfitting to surface cues (genre vocabulary, stylistic markers) that correlate with but are not identical to narrative structure. The taxonomy amplifies this: a soldier's "self-sacrifice in war" and a doctor's "altruistic martyrdom" both map to `MARTYRDOM` after clustering, improving both the TripletLoss training signal and the discriminability of the final embeddings. The 57.9:1 outcome compression ratio (Table 2) confirms that narrative resolutions follow a small set of universal archetypes, producing a particularly dense training signal for that component.

Limitations. First, the individual pipeline steps are not evaluated in isolation, making it difficult to assess which components contribute to performance and where errors are introduced. Second, no exhaustive parameter search was performed: taxonomy distance thresholds (0.35 / 0.50 / 0.55), model choices, prompt designs, and fine-tuning hyperparameters were set by manual inspection, leaving substantial room for improvement. Third, the task organizers provided 1,900 synthetic training triples, which we did not incorporate; we wanted to avoid spending financial resources on mining data of uncertain quality and were uncertain whether synthetically generated narratives would faithfully reflect the annotation criteria. Fourth, overly broad canonical action labels lose discriminative power, as detailed in the Error Analysis below.

Error Analysis. Manual inspection of 10 development stories (comparing initial profiles with standardized profiles) reveals two systematic failure patterns.

A. Generic canonical action labels lose discriminative power. The canonical action label `AUTHORITY_FIGURE_IDENTIFIES_PROCEDURAL_OVERSIGHT_OR_MISSING_NARRATIVE` appears in 5 of the 10 inspected stories spanning completely different genres: climate fiction, a 1960s youth drama, a wartime comedy, a crime caper, and a heist film. Similarly, `BOND_OF_TRUST_AND_MUTUAL_RESPECT_DEVELOPS_BETWEEN` appears in 4 of the 10 stories. When canonical labels subsume such diverse narrative events across entirely different genres, they cease to function as discriminative features, suggesting the taxonomy may over-cluster action terms.

B. Semantic inversion in action mapping. For a vampire-hunting story, the initial action "Protagonists observe a pattern of mysterious nocturnal attacks" is mapped by the taxonomy to `ANTAGONISTS_ARRIVE_SEEKING_ILLICIT_WEALTH_THROUGH_COERCION`—a label that describes the antagonists' role rather than the protagonists'. Agglomerative clustering can conflate semantically opposed events when their surface formulations share enough vocabulary (here, both involve external agents arriving with hostile intent), resulting in a mapping that inverts the narrative perspective.

7 Conclusion

We described our system for SemEval-2026 Task 4. The **Decompose & Align Cycle** extracts structured NarrativeProfiles, builds a controlled NarrativeTaxonomy, and fine-tunes an embedding model on profile representations via contrastive learning. Our Track A system (Gemini 3 Pro with the organizers' default prompt) achieves 69.50% (rank 16/44), above the GPT-4o-mini baseline. Our Track B system achieves 61.50% (rank 18/27), above the generic sentence-embedding baseline but below the

narrative-specific story-emb baseline. Key limitations include the absence of step-level evaluation, no hyperparameter search, and over-broad canonical action labels that reduce discriminative power. Incorporating the task's 1,900 synthetic triples is a possible direction for improvement, alongside replacing the clustering routine, with an agentic LLM workflow to generate candidates for the taxonomy.

Acknowledgments

We thank the SemEval-2026 Task 4 organizers.

References

Ludwig Berger. 1931. Le petit café (The Little Cafe). Paramount Pictures. French-language version of *Playboy of Paris* (1930), based on the play by Tristan Bernard. Cast: Maurice Chevalier, Yvonne Vallée, Tania Fédor.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: efficient finetuning of quantized llms. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

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.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *Preprint*, arXiv:2506.05176.