COGNAC at SemEval-2026 Task 4: Evaluating Narrative Components with LLMs for Hard Story Similarity Cases

Tisa Islam Erana*, Azwad Anjum Islam*, Anshu Kiran Sharma* & Mark A. Finlayson
Florida International University, Miami, FL, USA
{tisla016, aisla028, ashar076, markaf}@fiu.edu
*Authors share equal contribution

Abstract

We describe our system for the Narrative Similarity task at SemEval-2026 (Task 4), where the goal is to determine which of two candidate stories is more similar to an anchor story directly (Track A) or via vector representations (Track B). For Track A, our strategy leverages commercial, closed-source Large Language Models (LLMs) to generate multiple independent judgments per story triple. Simple majority voting provides strong performance in high-agreement cases, but it is unreliable when the judgments exhibit weak agreement. For difficult cases, we compare the stories along three narrative dimensions—theme, course of action, and outcome—prompting the LLMs to score similarity for each component on a scale of 1–4 and learning optimal combination weights on development data. We further find that chain-of-thought–style prompting with detailed reasoning outputs achieves comparable results to the scoring approach on difficult examples. We also conduct a data analysis revealing substantial annotation variability, which helps understand the difficulty of the task. Our system ranked 1^{st} in both tracks, achieving 0.78 test accuracy in Track A and 0.72 in Track B, where embedding only the course-of-action component yielded the best result.

1 Introduction

Measuring narrative similarity is a long-standing problem in computational linguistics and computational narratology. Similarity between stories, despite substantial differences in surface form, may derive from (i) shared themes and motivations (e.g., forbidden, tragic love story in *Romeo and Juliet* and *Pyramus and Thisbe*), (ii) analogous event sequences (e.g., stories sharing common narrative arcs such as a hero's journey to adventure and return), or (iii) comparable outcomes (e.g., fairy tales sharing happily-ever-after resolutions). SemEval-2026 Task 4 on narrative similarity (Hatzel et al., 2026) uses these three components to define similarity between stories and evaluates participating systems in two tracks: in Track A, systems are given an anchor story and two candidate stories and must determine which candidate is more similar to the anchor, while in Track B systems must produce vector representations of stories that align with their perceived similarity.

Our strategy for Track A followed a two-stage design. First, we obtained 7 independent LLM judgments for each triple (anchor, A, B) and determined the answer based on majority voting. This approach proved effective when the judgments exhibited strong agreement (7–0 or 6–1), but was significantly less reliable in low-consensus cases (4–3 or 5–2). We routed these difficult instances to the second stage.

In the second stage, we focused on the three components individually. We employed a component-wise scoring mechanism by first extracting the theme, course of action, and outcomes of each story, then scoring each candidate against the anchor on a scale of 1–4 for the three components using LLMs. We aggregated these component-wise scores using weights learned on development data and selected the candidate with the highest score. As an alternative approach, we used a Chain-of-Thought (CoT) style prompt to guide the LLMs to output structured comparisons of each candidate story to the anchor along each component, choose a component-wise winner, and then make a final decision. Both designs improved performance in difficult examples where majority voting falls short, but underperformed when applied to all cases. The two-stage system using either the component-scoring mechanism or CoT prompting for difficult cases and majority voting for others achieved 0.78 test accuracy on Track A, placing 1^{st} on the leaderboard.

We also experimented with additional methods such as normalizing stories by removing named en-

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 2290–2300

tities, RAG-style prompting, Propp-based function extraction etc., but these proved suboptimal for the task. To better understand the failure patterns, we conducted focused data analysis and observed that independent human re-annotations often diverged from the released labels, that gold annotations align more closely with LLM-generated judgments, and that in certain triples candidates appear more similar to each other than to the anchor.

For Track B, we explored whether component-focused representations better capture the similarity notion emphasized by the task. We produced vector representations of the theme, course of action, and outcomes of each story using a Gemini embedding model (Google, 2025b). Comparing these vectors both individually and in combination, we found that embeddings derived from the *course of action* component alone consistently yielded the strongest performance, placing 1^{st} on the Track B leaderboard with a score of 0.72.

The remainder of the paper is structured as follows. We provide background on narrative similarity and LLM-based solutions (§2), describe the task data (§3), and present our methodology, setup, and results for both tracks (§4, §5). We then discuss our own annotation study, error analysis, and other suboptimal approaches (§6), and conclude with contributions and limitations (§7, §8).

2 Related Work

Research on narrative similarity largely comes from narrative structure modeling, where stories are represented via events and their relations rather than just the text. Early unsupervised approaches introduced narrative event chains and schemas (Chambers and Jurafsky, 2008, 2009, 2010), later extended to richer schema and graph-based representations (Pichotta and Mooney, 2014; Li et al., 2018). Parallel efforts emphasized commonsense reasoning through benchmarks such as ROCStories and Story Cloze test (Mostafazadeh et al., 2016). More recent work has shifted towards neural representations that directly encode story-level semantics, including LLM-based embeddings trained on reformulations of same story (Hatzel and Biemann, 2024a) and benchmarks defining similarity via narrative elements such as characters, plot, setting, and theme (Chun, 2024).

Multi-sample inference improves language model performance by generating multiple candidate outputs and aggregating or selecting among them, an idea rooted in classical ensemble learning (Hansen and Salamon, 1990; Dietterich, 2000). This strategy has been widely applied in LLM reasoning tasks (Wang et al., 2023) including verifier-based selection approaches (Cobbe et al., 2021) and more recent work studying scaling and stability (Chow et al., 2025; Kang et al., 2025; Rakhsha et al., 2025) as well as applied fields (Garikipati et al., 2024; Huang et al., 2024).

Chain-of-Thought (CoT) prompting elicits intermediate reasoning steps before producing a final answer and has been shown to substantially improve performance across reasoning tasks (Wei et al., 2022; Kojima et al., 2022). Subsequent work has explored structured reasoning strategies, such as decomposing problems into ordered subproblems and tree or graph-based deliberation (Zhou et al., 2023; Yao et al., 2023; Besta et al., 2024).

3 Data

The narrative similarity shared task dataset comprises short movies and book summaries in English from Wikipedia. These stories were sourced from the Tell-Me-Again dataset (Hatzel and Biemann, 2024b) and range from 4 to 8 sentences in length. The organizers provided a sample set (39 labeled triples), a development set (200 labeled triples), and a test set (400 unlabeled triples). In Track A, each labeled instance contains one anchor and two candidate stories, and a label indicating the more similar candidate. Track B contains the same set of stories that appear in Track A as individual stories instead of triples. Systems must determine the more similar candidate directly in Track A, and produce vector representations of individual stories that reflect the underlying similarity relationships in Track B. Similarity is defined by three core narrative components: (1) the abstract themes, (2) the course of action, and (3) the outcomes of the story.

4 Track A Approach and Results

4.1 Majority Voting

For Track A, we first used a majority-voting scheme over seven (7) independent judgments per triple. We experimented with two off-the-shelf commercial models: `gpt-4.1-mini` (OpenAI, 2025) and `gemini-2.5-flash` (Google, 2025a). For each model, we used a minimal similarity prompt (Appendix A) at temperature 1.0 to encourage diverse, non-deterministic responses, then took the simple majority choice between A and B. Com-

Vote Diff.	gemini-2.5-flash		gpt-4.1-mini	
	Count	(%)	Count	(%)
1	17	8.5	8	4.0
3	23	11.5	11	5.5
5	28	14.0	12	6.0
7	132	66.0	169	84.5

Table 1: Vote-difference distributions (development set)

pared to single-shot baselines (one judgment per triple), majority voting improved development accuracy from 0.68 to 0.71 for `gpt-4.1-mini` and from 0.75 to 0.80 for `gemini-2.5-flash`.

We used the vote margin as a difficulty signal. For vote differences of 5 or 7 (6–1, 7–0), we accepted the majority decision as the final answer, while differences of 1 or 3 (4–3, 5–2) triggered second-stage processing (Section 4.2, 4.3). In the development set, such cases comprise 20% for `gemini-2.5-flash` and 9.5% for `gpt-4.1-mini` (Table 1). Due to `gemini-2.5-flash`'s consistently higher performance over the GPT model, we used its vote margins to define the difficult (20%) and easy (80%) data subsets.

4.2 Component-Scoring

For difficult triples identified in Section 4.1, our first approach applies a component-wise scoring mechanism over the three narrative similarity components: theme, course of action (CoA), and outcomes. We first performed a single extraction pass over all stories using `gpt-4.1-mini` to generate the three components. (Appendix B). Given these component representations, we compared the similarity of individual components of stories A and B against the anchor story for each triple, where the prompt (Appendix C) contained only the relevant component text and not the full stories.

We conducted this experiment with the same GPT and Gemini models as in Section 4.1. For each model, we generated seven responses per comparison, with each response containing a similarity score in the range 1–4 and a brief justification. We defined the similarity scores as (1) no similarity, (2) mostly unrelated with at least one significant similar aspect, (3) mostly similar with at least one significant dissimilar aspect, (4) essentially the same. We then averaged these seven scores to obtain a single score $S_{c,k}$ for each candidate $c \in \{A, B\}$ and component $k \in \{theme,\ coa,\ outcome\}$. We then aggregated the three component scores for each candidate story using a weighted average:

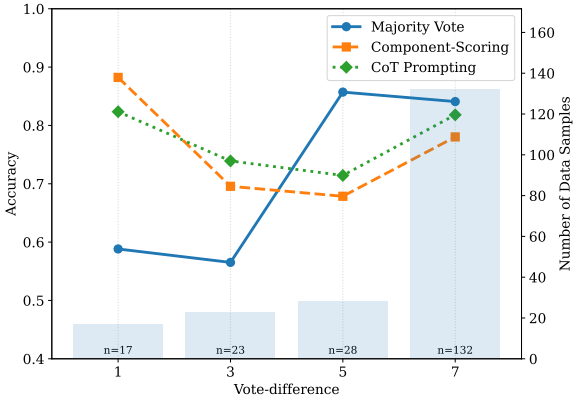

Figure 1: Accuracy of the three approaches for different vote-margins (development set)

$$S_c = \sum w_k \times S_{c,k}$$

To determine the optimal weights, we performed a grid search using the development set labels, enforcing $w_{theme} + w_{coa} + w_{outcome} = 1$. When the search space comprised the full development set, both `gemini-2.5-flash` and `gpt-4.1-mini` models converged on the same set of weights that maximizes model accuracy: $(w_{theme}, w_{coa}, w_{outcome}) = (0.3, 0.4, 0.3)$, which we used as the optimal weights for the test data. Notably, when the search space was restricted to difficult examples only, the Gemini model still found these same weights, while the GPT model identified optimal weights of $(0.5, 0.1, 0.4)$.

4.3 Chain-of-Thought (CoT) Prompting with Intermediate Reasoning

In our second approach for difficult triples, we designed a structured Chain-of-Thought (CoT) prompt (Appendix E) that decomposed narrative similarity into three equally weighted components: abstract theme, course of action, and outcomes. We defined abstract theme as the underlying motivations, moral framework, and central ideas; course of action as event sequence (setup → conflict → climax → resolution); and outcomes as the resolution or character fates. We experimented with two ways of wording the three narrative components in the prompt: a short version similar to the task definition, and a more detailed version that elaborated on what the model should attend to. While using the same components, in the detailed version we added simple guiding questions such as "what drives the story?" and "how do the events build up?". We found that the more detailed wording worked better on the difficult examples. The prompt guided the LLM to analyze each story in-

(Model) Approach	Easy (80%)	Difficult (20%)	Full (100%)
(gemini-2.5-flash)			
Majority Vote	**0.84**	0.58	0.80
(gpt-4.1-mini)			
Majority Vote	0.75	0.53	0.71
(gemini-2.5-flash)			
Scoring, Equal Weights	0.74	0.73	0.74
Scoring, Best Weights	0.76	**0.78**	0.77
(gpt-4.1-mini)			
Scoring, Equal Weights	0.74	0.65	0.72
Scoring, Best Weights	0.76	0.60	0.73
(gpt-4o)			
CoT Prompting	0.80	**0.78**	0.80
Two-stage			
(Majority + Scoring/CoT)	**0.84**	**0.78**	**0.83**
Test Set Performance			
Majority + Scoring	–	–	0.77
Majority + CoT	–	–	**0.78**

Table 2: Track A accuracy of different approaches and models on development and test data

dependently, extract its components, compare each candidate story to the anchor along each component, select a component-wise winner, and then produce a final similarity judgment. The model was also required to justify each intermediate and final decision. We collected seven independent judgments per triple per LLM and selected the final label by simple majority.

The structured output requirement of this experiment resulted in only 0.68 accuracy on difficult development examples with the `gpt-4.1-mini` model. Switching to a more capable `gpt-4o` (OpenAI, 2024a) improved accuracy to 0.78, which is the same as the component-scoring mechanism (§4.2) but at significantly higher cost and slower inference. Notably, CoT outperformed component-scoring on the full development set (Table 2).

4.4 Track A Results

Figure 1 illustrates how our three core approaches (§4.1, §4.2, §4.3) performed across different vote margins on development data. We see that the simple majority voting approach using a baseline prompt outperformed both of the more sophisticated component-scoring and CoT approaches for easy examples, but underperformed significantly for difficult examples. In contrast, both component-scoring and CoT maintained strong performance (0.78 accuracy) on difficult examples, with CoT demonstrating superior overall consistency.

Combining either of the component-scoring (by `gemini-2.5-flash`) or CoT (by `gpt4o`) approaches for difficult cases with majority voting (by `gemini-2.5-flash`) for easy cases produced equivalent development set performance. We submitted both of these two-stage combinations for the test set, where the CoT approach achieved marginally superior accuracy (0.78) over component-scoring (0.77), securing 1^{st} place on the Track A leaderboard.

Strategies	`miniLM`	`text-emb-3`	`gemini-emb`
Full text	0.55	0.66	0.65
Theme	0.61	0.62	0.60
CoA	0.61	0.65	**0.67**
Outcomes	0.54	0.64	0.64
Average	0.61	0.64	0.66
Weighted Avg	0.61	0.64	0.66
Test (CoA)	–	–	**0.72**

Table 3: Track B accuracy of different strategies and models on development and test data

5 Track B Approach and Results

For Track B, systems must produce vector representations of individual stories such that their cosine similarities align with the underlying similarity relations. We extracted the theme, course of action (CoA), and outcomes components of each story as described in Section 4.2. We then separately embedded the full story, and also embedded each component individually, computing their unweighted and weighted averages (using weights 0.3/0.4/0.3 from Section 4.2), for a total of six different embedding strategies. We evaluated three embedding models: `all-MiniLM-L6-v2` (SentenceTransformer baseline) (Wang et al., 2020; Reimers and Gurevych, 2019), `text-embedding-3-large` (OpenAI, 2024b), and `gemini-embedding-001` (Google, 2025b). Track B results are shown in Table 3. We see that CoA embeddings produced the best accuracy across all three models. This is consistent with Track A's finding, where the grid search also assigned the highest weight to the course-of-action component. So, we submitted CoA embeddings using the best performing model `gemini-embedding-001` for the test set, which produced an accuracy score of 0.72, placing 1^{st} on the Track B leaderboard.

6 Discussion

6.1 Data Analysis

During the preliminary experiments on the development data, we observed consistently low predictive performance. Manual inspection of the dataset revealed frequent disagreement between our interpretation of the annotation guidelines and the provided gold labels, as well as triples in which the candidate narratives appeared to be much more similar to each other than to the anchor (examples in Appendix D). We thus conducted a re-annotation of the sample set (39 triples) following the official annotation guidelines, where the second and third authors independently labeled each triple, with disagreements resolved through adjudication by the first author. We also produced LLM-generated labels using `gpt-4o` (OpenAI, 2024a), providing the task guidelines. The agreement in terms of Krippendorff's alpha between the gold and our adjudicated labels was $\alpha = 0.313$, while between gold and the LLM-generated labels was $\alpha = 0.406$. These values match the pre-adjudication inter-annotator agreement ($\alpha = 0.33$) reported in Hatzel et al. (2026). This reflects the dataset's inherent ambiguous nature and suggests vulnerability to annotator bias. Interestingly, the agreement between gold and LLM-generated labels exceeded agreement between gold and human re-annotations, which suggests that the gold annotations may reflect a particular interpretation of the guidelines that is more consistently reproduced by LLMs than by independent readers. Taken together, these findings indicate that annotation variability is a non-negligible factor when interpreting model performance on this dataset.

6.2 Error Analysis

In the development data, the majority voting approach failed on 25 easy examples, while the component-scoring and CoT approaches each failed on 9 difficult examples. Nearly all errors stem from ambiguous cases in which models inconsistently traded off high-level thematic overlap (e.g., gang conflicts, rebellion) against specific plot elements (e.g., accidental death, shipwreck). This often resulted in conflicting votes and near-tied overall scores. As discussed in §6.1, weak inter-annotator agreement in human labels also indicates that many instances exhibit inherent ambiguity and plausibly support multiple similarity judgments.

6.3 Other Sub-optimal Approaches

Removing named entities: The task annotation guidelines explicitly ignore character names and locations in deciding story similarity. We thus experimented with replacing the named entities in the stories with generic placeholders using an LLM to prevent distraction from surface details. However, this preprocessing step did not improve development accuracy (0.74) compared to baseline (0.80).

RAG-style few-shot prompting: We explored retrieval-augmented few-shot prompting (Islam et al., 2025) for Track A by retrieving 1–4 most similar labeled examples per triple. Similarity was computed via cosine distance between anchor stories using `gemini-embedding-001` embeddings of four separate representations: full text, theme, course of action, and outcomes. This approach also did not improve development accuracy (0.77) compared to baseline (0.80).

Propp-based function extraction: Motivated by Propp's narrative framework of 31 functions and 7 character roles (Propp, 1968), we explored three approaches: (i) direct function sequence matching (each story described as an ordered function chain), (ii) role-based scoring (each story described as a distribution over the seven roles), and (iii) a hybrid that combines Propp function similarity with sentence embedding similarity via adaptive weighting. These methods reached 0.49, 0.53, 0.63 accuracy on the development set, respectively, which suggests that Propp's framework transfers poorly to the Wikipedia story summaries in this dataset.

Multi-metric tie breaking: In this approach, we handled the difficult examples using a multi-metric semantic similarity module over `all-MiniLM-L6-v2` embeddings, combining optimal transport (Cuturi, 2013), Hungarian matching (Kuhn, 1955), sequence alignment (Needleman and Wunsch, 1970), and cosine similarity (Salton, 1989) with grid-search optimized weights. This approach achieved 0.67 accuracy on the development set.

7 Conclusion

Our two-stage hybrid system ranked 1^{st} on SemEval-2026 Task 4 Track A (0.78 accuracy) and Track B (0.72 accuracy). We showed that decomposing narratives into their core components—theme, course of action, and outcome—significantly improves LLMs' narrative similarity judgments in hard cases. By routing

low-agreement cases from a majority voting setup to component-wise scoring or structured CoT prompting, we achieved superior overall performance. Finally, our re-annotation study revealed substantial dataset ambiguity, underscoring the inherent subjectivity of the task and the importance of considering annotation variability when interpreting results.

8 Limitations

Our system relies exclusively on closed-source commercial LLMs, thus scalability is limited by cost considerations. Exploring whether fine-tuned open-source models can achieve comparable performance remains an important direction for future work. We also assumed that the extracted narrative components (theme, course of action, and outcome) are accurate but did not conduct any systematic evaluation beyond manual spot checks. While the manual checks did not reveal any significant failure pattern, the lack of formal evaluation means presence of such errors cannot be ruled out. Errors in this extraction step may therefore propagate to both our scoring and embedding methods. Finally, our Track B submission relies solely on course-of-action embeddings. While it proved effective, this may fail to capture higher-level thematic similarity, potentially overlooking narratives that are conceptually aligned but structurally different.

References

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michał Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and Torsten Hoefler. 2024. Graph of thoughts: solving elaborate problems with large language models. In *Proceedings of the Thirty-Eighth AAAI Conference on Artificial Intelligence and Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence and Fourteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'24/IAAI'24/EAAI'24. AAAI Press.

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.

Nathanael Chambers and Dan Jurafsky. 2009. Unsupervised learning of narrative schemas and their participants. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 2 - Volume 2*, ACL '09, page 602–610, USA. Association for Computational Linguistics.

Nathanael Chambers and Dan Jurafsky. 2010. A database of narrative schemas. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Yinlam Chow, Guy Tennenholtz, Izzeddin Gur, Vincent Zhuang, Bo Dai, Sridhar Thiagarajan, Craig Boutilier, Rishabh Agarwal, Aviral Kumar, and Aleksandra Faust. 2025. Inference-aware fine-tuning for best-of-n sampling in large language models. *Preprint*, arXiv:2412.15287.

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

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *Preprint*, arXiv:2110.14168.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transport. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Thomas G Dietterich. 2000. Ensemble methods in machine learning. In *International workshop on multiple classifier systems*, pages 1–15. Springer.

Anurag Garikipati, Jenish Maharjan, Navan Preet Singh, Leo Cyrus, Mayank Sharma, Madalina Ciobanu, Gina Barnes, Qingqing Mao, and Ritankar Das. 2024. OpenmedLM: Prompt engineering can out-perform fine-tuning in medical question-answering with open-source large language models. In *AAAI 2024 Spring Symposium on Clinical Foundation Models*.

Google. 2025a. Gemini 2.5 flash. `https://docs.cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-5-flash`. Model: gemini-2.5-flash, Accessed: 2026.

Google. 2025b. Gemini api documentation: Embeddings. `https://ai.google.dev/gemini-api/docs/embeddings`. Model: gemini-embedding-001, Accessed: 2026.

L.K. Hansen and P. Salamon. 1990. Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10):993–1001.

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, California. Association for Computational Linguistics.

Hans Ole Hatzel and Chris Biemann. 2024a. 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.

Hans Ole Hatzel and Chris Biemann. 2024b. Tell me again! a large-scale dataset of multiple summaries for the same story. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15732–15741, Torino, Italia. ELRA and ICCL.

Baizhou Huang, Shuai Lu, Xiaojun Wan, and Nan Duan. 2024. Enhancing large language models in coding through multi-perspective self-consistency. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1429–1450, Bangkok, Thailand. Association for Computational Linguistics.

Azwad Anjum Islam, Tisa Islam Erana, and Mark A. Finlayson. 2025. COGNAC at CQs-gen 2025: Generating critical questions with LLM-assisted prompting and multiple RAG variants. In *Proceedings of the 12th Argument mining Workshop*, pages 340–348, Vienna, Austria. Association for Computational Linguistics.

Zhewei Kang, Xuandong Zhao, and Dawn Song. 2025. Scalable best-of-n selection for large language models via self-certainty. *Preprint*, arXiv:2502.18581.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Harold W Kuhn. 1955. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97.

Zhongyang Li, Xiao Ding, and Ting Liu. 2018. Constructing narrative event evolutionary graph for script event prediction. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, IJCAI'18, page 4201–4207. AAAI Press.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

Saul B Needleman and Christian D Wunsch. 1970. A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of molecular biology*, 48(3):443–453.

OpenAI. 2024a. Introducing gpt-4o. https://openai.com/index/hello-gpt-4o/. Model: gpt-4o, Accessed: 2026.

OpenAI. 2024b. Openai api documentation: text-embedding-3-large. https://developers.openai.com/api/docs/models/text-embedding-3-large. Model: text-embedding-3-large, Accessed: 2026.

OpenAI. 2025. Introducing gpt-4.1 in the api. https://openai.com/index/gpt-4-1/. Model: gpt-4.1-mini, Accessed: 2026.

Karl Pichotta and Raymond Mooney. 2014. Statistical script learning with multi-argument events. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 220–229.

Vladimir Iakovlevich Propp. 1968. *Morphology of the folktale*. University of Texas Press.

Amin Rakhsha, Kanika Madan, Tianyu Zhang, Amir massoud Farahmand, and Amir Khasahmadi. 2025. Majority of the bests: Improving best-of-n via bootstrapping. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

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*. Association for Computational Linguistics.

Gerard Salton. 1989. Automatic text processing: The transformation, analysis, and retrieval of. *Reading: Addison-Wesley*, 169.

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 *Proceedings of the 34th International Conference on Neural Information Processing Systems*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L. Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: deliberate problem solving with large language models. In *Proceedings of the 37th International Conference on*

Neural Information Processing Systems, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc V Le, and Ed H. Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations*.

A Baseline Similarity Prompt

```
Given an anchor story and two options A and B, determine which one is a more similar story to the anchor. Provide your answer in a valid JSON format as following:
{answer: <a or b>}
```

B Prompt for Extracting Components

```
Describe a given story in JSON format. You need to describe the following three components:

1) Overall abstract theme: Describe in brief the central ideas, core motifs and defining constellation of problems. For example, in both these 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."
Theme: A story about people isolated from outside world in a journey, and how it affects their interpersonal relationship.

2) Course of action/events: Describe in brief the sequence of events that actually happens in the story. For example, in the following stories:
A: "After the ship capsizes and Alice barely makes it out alive, she starts living life to the fullest with a new-found perspective about how precious life is."
Events: Alice's ship capsizes. Alice barely makes it out alive. Alice starts living life to the fullest.
B: "Alex loses his engagement ring while swimming. He freaks out, and after hours of diving for it, he still cannot find it."
Events: Alex loses his engagement ring while swimming. Alex freaks out. Alex looks for it. Alex fails to find it.

3) The outcomes: Describe in brief the final ending or outcomes of the story. For example, in the following stories:
A: "Anna loses her purse. She retraces her steps but cannot find it. Dan finds it and helpfully returns it to her."
Outcome: Someone finds a lost item and returns to owner.
B: "Brian lost his backpack. He was terrified because there were important documents in it. After an hour of intense search he finally found it."
Outcome: Someone finds their lost item.
C: "Jill was driving home when another car suddenly crashed into hers. After receiving medical attention, she recovered within just days and now advocates for traffic safety."
Outcome: A person advocates for traffic safety after recovering from car crash.
You should produce a valid JSON object with the three attributes describing the given story: "theme", "events" and "outcome". Do not produce any extra explanation or additional text.
```

C Prompt for Similarity Score

```
Given an anchor text and two options: option a and option b. Determine how similar each option is to the anchor text on a scale of 1-4:

1: Unrelated / No similarity.
2: Mostly unrelated, but there is at least one significant aspect of the story that is similar to the anchor.
3: Mostly similar, but there is at least one significant aspect that is different from the anchor.
4: Essentially the same.
Provide justification for your answer. Provide your answer in a valid JSON format with the following attributes:

{ "a_similarity_score": <number between 1-4>,
"a_reason": <justification for text a's score>,
"b_similarity_score": <number between 1-4>,
"b_reason": <justification for text b's score>
}
```

D Data Examples

Both examples are extracted from the provided sample dataset.

D.1 Example 1

Example where the independent annotators (authors) agree with each other but differ from the gold and LLM annotations.

Anchor story: *"In an artist's studio, rich Parisian art dealer Félicien Mézeray sees the old soldier Legrain, whose back has a tattoo by Modigliani. This he sells unseen to two American dealers and the rest of the film revolves around his efforts to literally get the skin off Legrain's back. The price Legrain wants is the restoration of his old family home in the country, which turns out to be the huge crumbling castle of Paluel in remote Périgord, while he turns out to be the last and extremely eccentric Count of Montignac. The plot bears a very strong resemblance to Saki's short story The Background."*

Story A: *"Papa Gimplewart (Davidson) exchanges his house, in order to escape the antics of inmates of the lunatic asylum next door, including characters played by Laurel and Hardy. Unfortunately, the new house turns out to be 'Jerry-built', put up in two days. After several disasters occur, Papa Gimplewart asks "Is there anything else can happen?". He then realizes that the inmates from the asylum have just moved in next door. Among the disasters are a mop removing the color from the kitchen floor;dirty bath water leaking down from*

upstairs and into the communal coffeepot;and a piano sliding on an uneven floor that crashes through a wall and demolishes the family car. Excerpts from this film appeared in the Robert Youngson documentary LAUREL AND HARDY'S LAUGHING 20's(1965)"

Story B: *"1940 The Great Depression is over and World War II had just begun. King of the con men Fargo Gondorff is released from prison and reassembles his cronies for another con, out to avenge the murder of his lifelong pal and fellow con artist Kid Colors who was kidnapped, beaten, and then shot. Gondorff's young protege Jake Hooker attempts to pull a scam on wealthy "Countess Veronique," who instead pulls one on him and turns out to be a grifter herself named Veronica. Coming up with a boxing con, Gondorff's goal is to sting both Lonnegan, the notorious banker and gangster who wants revenge from a previous con, and Gus Macalinski, a wealthy local racketeer. One or both of them is behind Kid Colors' death. Hooker pretends to be a boxer who is about to throw a big fight. Macalinski is not only hoodwinked into losing hundreds of thousands of dollars, but he is also talked into changing his original wager by Lonnegan. While one gangster takes care of the other, Gondorff and Hooker head for the train station with a bag full of money, tickets out of town and a final twist from Veronica."*

Gold: *Story B*.

D.2 Example 2

Example where story A and B exhibit stronger similarity to each other than the anchor, based on the abstract theme of infidelity in both the candidates:

Anchor story: *"Sara is a teenager who lives with her family, who are Jehovah's Witnesses. The family's devout image is questioned when the parents divorce as a consequence of the father's infidelity. One night at a party Sara meets Teis, an older boy who takes an interest in her. Teis is not a Witness, and their relationship is rejected by her father, but Sara falls in love and begins to doubt her faith. Facing ostracism from her faith and family, Sara must make the toughest choice of her young life."*

Story A: *"In 1931 in Paris, France, Anaïs Nin is in a stable relationship with her husband Hugo, but longs for more out of life. When Nin first meets Henry Miller, he is working on his first novel. Nin is drawn to Miller and his wife June, as well as their bohemian lifestyle. Nin becomes involved in the couple's tormented relationship, having an affair with Miller and also pursuing June. Ultimately, Nin helps Miller to publish his novel Tropic of Cancer, but catalyzes the Millers' separation, while she returns to Hugo."*

Story B: *"Leading chemist Hans Schramm is betrothed to Hanna, but falls in love with her younger sister Franka. The two attempt to repress their feelings, but eventually begin an affair. When Hans is extorted by a group of West German agents, who demand to know about his secret work, he is gripped by panic and decides the only way out is to flee to the West. Franka discovers his plans and informs the Stasi. Hans perceives it as betrayal at first, but after all ends well, he realizes she only wanted the best for him."*

Gold: *Story A*.

E Chain-of-Thought Prompt

```
You are an expert in narrative analysis. Compare two stories to determine which is MORE narratively
similar to an anchor story.
IMPORTANT: Narrative similarity has THREE components (all equally important):
1.Abstract Theme: Core ideas, character motivations, moral framework, underlying message
2.Course of Action: Sequence of key events, plot structure, turning points, character actions
3.Outcomes: How the story resolves, character fates, consequences, final state

ANCHOR STORY:
{anchor_text}

STORY A:
{text_a}

STORY B:
{text_b}

INSTRUCTIONS:
First, analyze each story by identifying:
1.The main theme/motivation (what drives the story? what's the core idea?)
2.The key event sequence (setup → conflict → climax → resolution)
3.The outcomes/ending (how does it resolve? what happens to characters?)

Then compare A and B to the ANCHOR on each dimension.

Output your analysis in JSON format:
{{
"anchor_analysis": {{
"theme": "brief description of core theme and motivations",
"key_events": ["event1", "event2", "event3"],
"outcomes": "how it ends and character fates"
}},
"story_a_analysis": {{
"theme": "brief description of core theme and motivations",
"key_events": ["event1", "event2", "event3"],
"outcomes": "how it ends and character fates"
}},
"story_b_analysis": {{ "theme": "brief description of core theme and motivations",
"key_events": ["event1", "event2", "event3"],
"outcomes": "how it ends and character fates"
}},
"comparison": {{
"theme_similarity": {{
"a_vs_anchor": "How similar are the themes and motivations?",
"b_vs_anchor": "How similar are the themes and motivations?",
"winner": "A or B - which has more similar theme?"
}},
"event_similarity": {{
"a_vs_anchor": "How similar are the event sequences and plot structure?",
"b_vs_anchor": "How similar are the event sequences and plot structure?",
"winner": "A or B - which has more similar events?"
}},
"outcomes_similarity": {{
"a_vs_anchor": "How similar are the endings and resolutions?",
"b_vs_anchor": "How similar are the endings and resolutions?",
"winner": "A or B - which has more similar outcomes?"
}}
}},
"final_decision": "A or B",
"confidence": "high/medium/low",
"reasoning": "1-2 sentence explanation of why this choice is more narratively similar overall"
}}
```