MoodMetric at SemEval-2026 Task 4: Dense Transformer Networks for Narrative Story Similarity and Representation

Bolisetty Samanvitha, Shreya Ashar, Nishchay Mittal, and Pruthwik Mishra
Sardar Vallabhbhai National Institute of Technology (SVNIT), Surat, India
{u24ai024, u24ai003, u24ai044, pruthwikmishra}@aid.svnit.ac.in

Abstract

Modeling semantic similarity between long-form narratives is substantially more challenging than sentence-level matching. The major bottlenecks arise due to continuity at the level of structural elements such as characters, entities, or events, causal dependencies, and implicit thematic coherence. In this work, we investigate transformer-based dense retrieval methods for the SemEval-2026 Task 4 narrative similarity challenge, focusing primarily on **Track A** (comparative narrative ranking) and **Track B** (narrative embedding generation). We evaluate multiple pretrained encoder architectures—including DeBERTa-v3, BGE-Base, BGE-Large, and E5-Large—adapted using triplet and contrastive metric learning objectives. Our study analyzes the effects of model scale, pooling strategy, layer freezing, training duration, and cross-validation ensembling on generalization performance. Across experiments on the **Track A pairwise ranking task**, we observe that larger contrastively pretrained embedding models consistently outperform smaller variants, but performance saturates rapidly given approximately 2,000 training triplets. Moderate fine-tuning (4–5 epochs) yields optimal Track A validation accuracy, while extended training leads to clear overfitting despite near-zero training loss. Instruction-tuned embeddings do not demonstrate significant advantages over contrastively aligned alternatives for this narrative task. Finally, arithmetic ensemble averaging of diverse embedding models produces the most robust Track A representations, achieving approximately **65.0% Track A validation accuracy**.

1 Introduction

Modeling semantic similarity between long-form narratives remains a challenging problem in natural language processing. Unlike sentence-level similarity or paraphrase detection, narrative comparison requires capturing event progression, implicit causality, character intent, and thematic coherence. Subtle differences in plot structure or temporal ordering can significantly alter meaning despite high lexical overlap, making surface-level similarity metrics insufficient for robust story-level comparison (Chun, 2024).

Recent advances in transformer-based encoders have improved semantic representation learning for retrieval and ranking tasks. Pretrained models such as BGE, E5, DeBERTa, and Sentence-BERT produce dense vector representations that capture contextual semantics beyond lexical similarity. When fine-tuned with contrastive or triplet-based objectives, these models can be adapted for narrative similarity. However, low-resource settings with limited training triplets introduce substantial overfitting risk and limit generalization.

In this work, we investigate transformer-based dense retrieval approaches for two tasks proposed by Hatzel et al. (2026): (1) **Track A**: comparative narrative ranking, a pairwise ranking task where the model determines which of two candidate stories is semantically closer to an anchor narrative; and (2) **Track B**: fixed-dimensional narrative embedding generation. We evaluate multiple pretrained encoders—including DeBERTa-v3, BGE-Base, BGE-Large, and E5-Large—under triplet and contrastive learning objectives, and analyze the effects of model scale, pooling strategy, layer freezing, training duration, and embedding-level ensembling.

2 Related Work

Early computational work on narrative modeling Chambers and Jurafsky (2008) proposed unsupervised methods for learning narrative event chains that capture typical event orderings. More recent studies investigate narrative similarity directly. Saldias and Roy (2020) showed that structural narrative features improve similarity modeling for spo-

Proceedings of the The 20th International Workshop on Semantic Evaluation (2026), pages 2664–2672

ken personal narratives, while Piper et al. (2021) highlighted the importance of incorporating narratological theory, including event sequencing, and character arcs into NLP models.

Recent advances in dense retrieval models based on pretrained transformers have significantly improved semantic representation learning. Contrastively pretrained embedding models such as BGE (Xiao et al., 2024) and E5 (Wang et al., 2024) achieve strong performance on semantic retrieval tasks, but their training data mainly consist of web documents and question-answer pairs rather than narrative texts. This motivates task-specific fine-tuning of dense encoders on narrative similarity data, which we explore in this work.

Specialized tasks in different Semantic Evaluation (SemEval) workshops have focused on narrative understanding and similarity, where similarity is evaluated based on abstract themes, courses of action, and story outcomes, often utilizing multilingual news or story datasets (Chen et al., 2022; Piskorski et al., 2025). The current SemEval Task4 [1] attempts to evaluate the narrative similarity in three core aspects of: (1) abstract themes of the story, (2) the course of action, and (3) the story outcomes.

3 Dataset

The dataset provided by the organizers consists of narrative triplets in JSONL format. Each sample contains an anchor narrative a, a positive narrative p, and a negative narrative n indicating relative semantic similarity. The details are shown in Table 1.

Track	Type	#Samples
A	Train	1900
	Dev	200
	Test	400
B	Train	1900
	Dev	200
	Test	849

Table 1: Task-wise Original Dataset Details

All empty or null entries were replaced with empty strings. For **Track A**, the task is formulated as a triplet ranking problem. Given (a, p, n), the model learns embeddings such that:

$$\cos(a, p) > \cos(a, n)$$

For **Track B**, models produce fixed-dimensional embeddings optimized for cosine-similarity-based retrieval.

3.1 Synthetic Data Augmentation

We use groq API [2] to generate additional synthetic training triplets. The models used for this task are:

- Llama 3.1 8B Instant (Grattafiori et al., 2024)
- GPT OSS 20B (Agarwal et al., 2025)
- Qwen 3 32B (Yang et al., 2025)
- Groq Compound combining GPT-OSS 120B (Agarwal et al., 2025) and Llama 4 (Adcock et al., 2026)

We generated 836 synthetic samples in this process. The prompt template for the synthetic data generation is provided in Table 5 under appendix A.1.

4 Experimental Setup

We use the HuggingFace framework (Wolf et al., 2020) for fine-tuning different encoder-only models for both tasks. We develop a logistic regression based model as a simple baseline where we concatenate sparse TF-IDF (Sparck Jones, 1972) and dense SBERT (Thakur et al., 2021) representations as features. We experiment with both CLS pooling and mean pooling from an encoder-only BERT (Devlin et al., 2019) model to represent a narrative. All final embeddings are L2-normalized prior to similarity computation. For track B, the representation of a story narrative is the mean of pooled outputs from different encoder only variants.

4.1 Training Objective

All models are trained using a triplet margin loss.

$$L = \max(0, \cos(a, n) - \cos(a, p) + m) \quad (1)$$

where m denotes the margin hyperparameter (typically 0.35).

Some experiments additionally incorporate contrastive softmax loss with temperature scaling.

$$L_{\text{cont}} = -\log \frac{e^{\cos(a,p)/\tau}}{e^{\cos(a,p)/\tau} + e^{\cos(a,n)/\tau}} \quad (2)$$

The total loss in hybrid experiments is:

$$L_{\text{total}} = L_{\text{margin}} + \alpha L_{\text{cont}}$$

[1] https://narrative-similarity-task.github.io/

[2] https://console.groq.com

4.2 Models For Track A

For **Track A** (pairwise narrative ranking), we fine-tune the following encoder-only models.

- DeBERTa-v3-base (184M parameters, 768-dim) (He et al., 2021)
- BGE-Base-en-v1.5 (110M parameters, 768-dim) (Xiao et al., 2024)
- BGE-Large-en-v1.5 (335M parameters, 1024-dim) (Xiao et al., 2024)
- E5-Large-v2 (335M parameters, 1024-dim) (Wang et al., 2024)

We experiment with several model-specific adjustments such as layer freezing, cross-validation ensemble, and margin variations. For Track B, final embeddings are computed via arithmetic averaging:

$$\mathbf{E}_{\text{ensemble}} = \text{Normalize}\left(\frac{1}{N}\sum_{i=1}^{N}\mathbf{E}_i\right) \quad (3)$$

This reduces model-specific bias and improves generalization.

We evaluate five modeling paradigms for comparative narrative similarity: (1) a lexical–semantic hybrid base classifier, (2) a task-adapted transformer (DeBERTa), (3) pretrained dense embedding models (BGE variants), (4) text embeddings with contrastive pretraining and weak supervision (E5) and (5) a multi-model embedding ensemble.

4.2.1 Base Classifier

We build a supervised hybrid similarity classifier that combines sparse TF-IDF features with dense SBERT embeddings. Texts are represented using unigram and bigram TF-IDF vectors (8,000 dimensions) and 384-dimensional SBERT embeddings. For each $(anchor, A, B)$ tuple, we concatenate the element-wise differences between the anchor and each candidate across both representations, forming a 16,768-dimensional feature vector. This vector is input to a logistic regression classifier (Berkson, 1944) to predict which candidate is semantically closer to the anchor.

4.2.2 DeBERTa-v3-Base

We adapt DeBERTa-v3-Base (12 layers, hidden size 768) to a Siamese ranking framework which generates a pairwise embedding for a pair of anchor-candidate. To reduce the positional bias, we swap candidate A and B with label inversion. First four encoder layers are frozen to preserve the pretrained linguistic structure. We use attention mask weighted mean pooling where the training loss is composed of objectives for masked language modeling and replaced token detection.

4.2.3 BGE-Base

BGE-Base (Xiao et al., 2024) is a 12-layer transformer with a hidden size of 768 and 768 dimensional embeddings pretrained via contrastive learning. We use a cosine-similarity triplet loss for anchor–positive–negative separation.

4.2.4 BGE-Large

BGE-Large (Xiao et al., 2024) is a 24-layer transformer with a hidden size of 1024 producing 1024 dimensional output embeddings, pretrained using hard negatives. This model shows strong zero-shot performance. We fine-tune the model for 4 epochs, further training causes the model to overfit with an increase in validation accuracy.

4.2.5 E5-Large-v2

E5-Large-v2 (Wang et al., 2024) is 24-layer transformer producing 1024-dimensional embeddings. It is pretrained with instruction-style prompts under contrastive objectives for semantic retrieval with weak supervision signals from heterogeneous text pairs. For Track A, we fine-tune the encoder using a cosine-based triplet ranking loss.

4.2.6 Ensemble Model

To reduce model-specific bias, we aggregate embeddings:

$$\mathbf{E}_{\text{ensemble}} = \text{Normalize}\left(\frac{1}{N}\sum_{i=1}^{N}\mathbf{E}_i\right)$$

Models included: BGE-Large, BGE-Base, DeBERTa, E5-Large, and additional BGE variants.

4.3 Models for Track B

Track B follows the dense embedding framework described in Track A, where each story is encoded using a transformer-based sentence encoder and optimized under a triplet ranking objective.

The primary distinction in Track B lies in generating high-quality story-level embeddings for retrieval rather than pairwise classification. Each

story embedding is L2-normalized to ensure stable cosine similarity comparisons. For the final submission, we ensemble embeddings from multiple independently trained models via arithmetic averaging followed by normalization, improving robustness and reducing model-specific bias.

For Track B, our final system employs a multi-faceted approach to narrative embedding and retrieval. We leverage an ensemble of **BGE-Large** (335M parameters) and **DeBERTa-v3-base** (183M parameters) models, each trained via 5-fold cross-validation to ensure robust generalization. The ensemble uses model averaging to compute L2-normalized 1024-dimensional embeddings, explicitly optimized for cosine-similarity-based retrieval and narrative matching tasks.

4.4 Fine-Tuning Details

Our training leverages a hybrid loss formulation combining **contrastive loss** (with temperature scaling $\tau = 0.05$ for numerical stability) and **margin ranking loss** (margin $= 0.4$), enabling the models to learn fine-grained similarities while maintaining ranking-aware separation between positive and negative narrative pairs. The BGE-Large variant was trained on an expanded dataset of 2,736 samples with tuned hyperparameters (batch size 12, learning rate 1.5×10^{-5}, 5 epochs), while DeBERTa-v3-base underwent supervised fine-tuning with layer freezing and early stopping (patience = 3) to prevent overfitting on the ranking task.

4.5 Key Design Decisions

- **Ensemble Strategy:** Averaging predictions from 5-fold models reduces variance and leverages diverse feature representations learned across different data splits.
- **L2 Normalization:** Enables efficient cosine similarity computation and provides interpretable embedding geometry aligned with retrieval objectives.
- **Mixed-Precision Training:** Used on CUDA to accelerate convergence while maintaining gradient stability.
- **Data Augmentation:** Pseudo-labeling on test data during pre-training increased effective training set size and improved domain coverage.

4.6 Performance

The final system achieves **65.5% accuracy** on the Track B test set, representing a substantial improvement over single-model baselines. Cross-validation analysis showed consistent performance (5 Fold validation accuracies: 79.17%–80.65% for DeBERTa-v3), demonstrating strong generalization across narrative subdomains.

4.7 Optimization Parameters

The parameters for optimizing the BERT models are presented in Table 2. For each model, the parameters are kept the same.

Optimizer	AdamW
Learning rate	1×10^{-5} to 2×10^{-5}
Batch size	8–16
Warmup ratio	5%
Mixed Precision (AMP)	Enabled
Hardware	NVIDIA T4/H100 GPUs

Table 2: Optimization Parameters For Transformer Models

5 Experimental Results

All results reported in this section correspond to **Track A**, the pairwise narrative ranking task, where the model predicts which of two candidate stories is semantically closer to a given anchor narrative.

Table 4 summarizes the **Track A validation accuracy** for all evaluated systems on the 200-sample development set. The hybrid TF-IDF + SBERT baseline gives a steady validation accuracy of 57.5%. Among the transformer-based models, larger models are visibly superior: BGE-Large is the best single model with a peak **Track A validation accuracy of 64.5%**. The ensemble of all models provides a marginal further improvement to **65.0% Track A validation accuracy**, though without a substantial gain over BGE-Large alone.

Table 4 reports the **official Track A test accuracy** obtained from CodaBench for each submitted configuration.

5.1 Ablation Study: Effect of Synthetic Data

To evaluate the impact of synthetic data augmentation, we conduct an ablation study comparing model performance with and without the additional generated triplets on the **Track A validation set**.
Observation: The inclusion of synthetic data improves Track A model performance by increasing training diversity. This helps the model generalize

Setting	Track A Val. Accuracy
Without synthetic	63.2%
With synthetic	65.0%

Table 3: Impact of synthetic data augmentation on Track A validation performance.

better to unseen narrative structures, especially in low-resource settings.

5.2 Synthetic Data Validation

We generated 836 synthetic triplets to augment the training data. To ensure data quality, we performed validation using both manual and automated checks.

Validation process:

- Verified semantic consistency between anchor and similar stories
- Ensured dissimilar stories differed in theme, events, or outcomes
- Checked narrative coherence and readability
- Removed duplicates and malformed samples

Outcome: The majority of synthetic samples were coherent and aligned with the task definition, making them suitable for training and contributing to improved Track A performance.

Across experiments, several trends emerge. The hybrid TF-IDF + SBERT baseline underperforms dense transformer models due to limited narrative modeling capacity and sensitivity to the small dataset. DeBERTa is affected by sequence length limits and shows strong cross-validation performance but poor held-out **Track A** accuracy, indicating overfitting.

BGE-Large consistently outperforms BGE-Base on **Track A**, highlighting the importance of model capacity and embedding dimensionality. Performance peaks around 4 epochs, while extended training leads to overfitting, suggesting data quantity as the primary bottleneck. E5-Large, despite instruction-tuned pretraining, shows no significant advantage over BGE-Large on the Track A ranking task, and ensemble averaging provides no complementary gains.

Overall, contrastively pretrained models with task-aligned objectives produce stronger Track A representations than hybrid or instruction-tuned approaches in low-resource narrative similarity settings.

5.3 Qualitative Error Analysis

We analyze a few failure cases to understand model limitations on the Track A pairwise ranking task.

Surface similarity confusion: The model prefers candidates with high lexical overlap (e.g., "athlete", "training") even when narrative outcomes differ (failure vs success).

Outcome mismatch: In several cases, the model selects stories with similar setups but different endings, indicating weak sensitivity to outcome alignment.

Summary: These errors suggest the model relies more on surface-level similarity than deeper narrative structure such as outcomes and implicit themes.

6 Key Observations

Several consistent findings emerged across our Track A experiments:

Moderate fine-tuning is optimal. Performance peaks at 4–5 training epochs for large models. Extended training causes overfitting given the limited dataset size (≈2.5k samples).

Model scale matters, but saturates quickly. Larger models (BGE-Large, E5-Large) consistently outperform smaller ones on Track A, but gains diminish rapidly beyond a certain scale under low-data conditions.

Contrastive pretraining shows marginal gains. E5-Large's contrastive pre-training improved training stability but did not yield significant Track A accuracy improvements compared to BGE-Large.

Ensemble averaging improves robustness. Combining diverse model architectures via arithmetic mean consistently reduces variance and improves Track A generalization over any single model.

Data quality dominates modest data scaling. The quality of triplet training examples has a greater impact on final Track A performance than small increases in dataset size.

7 Future Directions

While the ensemble approach provides solid performance, we identify several avenues for improvement:

Model	Params	Emb. Dim	Test Acc.	Val Acc.	Notes
SBERT + TF-IDF	22M	384 + sparse	59.0%	57.5%	Baseline
DeBERTa-v3-base	183M	768	52.0%	79.16%	Siamese fine-tuned
BGE-Base	110M	768	59.5%	58.5%	Efficient
BGE-Large	335M	1024	**65%**	64.5%	Best single model
E5-Large	335M	1024	62.0%	60.0%	Zero-shot + FT
BGE-Large (Min. Ranking)	335M	1024	62.0%	63.5%	Ranking baseline
BGE-Large (FT, 4 ep.)	335M	1024	65%	64.5%	Improved training
Ensemble	—	1024	**XX.X%**	**61.0%**	—

Table 4: Comprehensive comparison of all models on Track A, including validation and official test accuracy. *The Ensemble test accuracy is marked as XX.X% as this configuration was not submitted for official evaluation. Min. = Minimal, FT = Fine-tuned, Acc. = Accuracy.*

- **Harder Negative Mining:** Implementing curriculum learning or online hard negative selection could strengthen the learned embedding space by focusing on challenging narrative pairs.
- **Cross-Encoder Re-ranking:** A learned cross-encoder could refine top-k retrieval results using fine-grained pairwise comparisons.
- **Domain-Adaptive Pre-training:** Continued pre-training on narrative-specific corpora before fine-tuning could improve task transfer and capture genre-specific narrative structures.
- **Semantic Data Augmentation:** Back-translation, paraphrasing, or synthetic narrative generation could expand training diversity.
- **Multi-Task Learning:** Joint training on Track A ranking and Track B retrieval might improve representation quality through shared semantic knowledge.

8 Reproducibility and Resources

Code and models are publicly available. Code: https://github.com/samanvitha7/SemEval2026-task4 (Bolisetty et al., 2026). Models: DeBERTa (https://huggingface.co/samanvitha7/semeval-hcp-deberta), E5-Large (https://huggingface.co/samanvitha7/semeval2026-e5_large-checkpoints), and BGE-Large variants (https://huggingface.co/samanvitha7).

These checkpoints correspond to different training strategies explored in our experiments, including data augmentation, layer freezing, and extended fine-tuning.

9 Conclusion

We have described our system for SemEval-2026 Task 4, covering both **Track A** (comparative narrative ranking) and **Track B** (narrative story embedding generation). For **Track A**, we formulate the task as a pairwise ranking problem where the model determines which of two candidate stories is semantically closer to an anchor narrative. Through systematic exploration of transformer-based encoders under triplet supervision, we find that BGE-Large-en-v1.5 provides the strongest standalone **Track A validation accuracy of 64.5% and on submission the test accuracy is 65%** (25^{th} $position$)**.** For **Track B**, our final system produces L2-normalized 1024-dimensional embeddings via ensemble averaging, optimized for cosine-similarity-based retrieval with validation accuracy of 65% and test accuracy 65.5% (16^{th} $position$). Future work could explore harder negative mining strategies, cross-encoder re-ranking, or domain-adaptive pre-training to further improve both narrative ranking and embedding quality.

Limitations

We did not use any decoder-only model for this task. We could only implement 4 BERT variants with different pretraining objectives that limits the coverage of our experiments. We augment only $\approx$ 800 synthetic triples to our training data. Increasing the number of synthetic samples could have increased the performance of the submitted models.

Acknowledgments

We thank SVNIT Surat for the Aurora High-Performance Computing HPC Cluster and the GPU resources provided to carry out our experiments.

References

Aaron Adcock, Aayushi Srivastava, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pande, Abhinav Pandey, Abhinav Sharma, Abhishek Kadian, Abhishek Kumawat, Adam Kelsey, and 1 others. 2026. The llama 4 herd: Architecture, training, evaluation, and deployment notes. *arXiv preprint arXiv:2601.11659*.

Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, and 1 others. 2025. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*.

Joseph Berkson. 1944. Application of the logistic function to bio-assay. *Journal of the American statistical association*, 39(227):357–365.

Samanvitha Bolisetty, Shreya Ashar, Nishchay Mittal, and Pruthwik Mishra. 2026. Semeval-2026 task 4: Narrative similarity codebase. `https://github.com/samanvitha7/SemEval2026-task4`. GitHub repository.

Nathanael Chambers and Dan Jurafsky. 2008. Unsupervised learning of narrative event chains. In *Proceedings of ACL-08: HLT*, pages 789–797.

Xi Chen, Ali Zeynali, Chico Camargo, Fabian Flöck, Devin Gaffney, Przemyslaw Grabowicz, Scott A. Hale, David Jurgens, and Mattia Samory. 2022. SemEval-2022 task 8: Multilingual news article similarity. In *Proceedings of the 16th International Workshop on Semantic Evaluation (SemEval-2022)*, pages 1094–1106, Seattle, United States. Association for Computational Linguistics.

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.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing. *Preprint*, arXiv:2111.09543.

Andrew Piper, Richard Jean So, and David Bamman. 2021. Narrative theory for computational narrative understanding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 298–311, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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

Belen Saldias and Deb Roy. 2020. Exploring aspects of similarity between spoken personal narratives by disentangling them into narrative clause types. In *Proceedings of the First Joint Workshop on Narrative Understanding, Storylines, and Events*, pages 78–86, Online. Association for Computational Linguistics.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

Nandan Thakur, Nils Reimers, Johannes Daxenberger, and Iryna Gurevych. 2021. Augmented SBERT: Data augmentation method for improving bi-encoders for pairwise sentence scoring tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 296–310, Online. Association for Computational Linguistics.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024. Text embeddings by weakly-supervised contrastive pre-training. *Preprint*, arXiv:2212.03533.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven

Le Scao, Sylvain Gugger, and 3 others. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. *Preprint*, arXiv:2309.07597.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

A Appendix

A.1 Prompt Template for Synthetic Data Creation

Prompt Template

You are an expert who can generate a similar story and dissimilar story given an input anchor story.
- For a given anchor story, generate only a single similar story and only a single dissimilar story.
- Do not generate any additional anchor texts other than the existing ones in the input CSV file.
- For multiple anchor stories generate similar and dissimilar stories for each anchor story.
- Save each line in JSON format corresponding to an anchor story, the similar story, and the dissimilar story.
- Do not include any explanations or extra text.
- Do not generate code or other kinds of textual noise.
- For thinking or reasoning, do it internally without including it in the output.
- Ensure that the similar story and dissimilar story are coherent and contextually relevant.
- The similar story should have similar themes, settings, and character types as the anchor story.
- The dissimilar story should have different themes, settings, and character types compared to the anchor story.
- The length of each similar story and each dissimilar story should be approximately the same as the corresponding anchor story.
- The output format should be as follows: For n anchor stories:

```
{
 {
  "anchor_story": "<anchor_story_text_1>",
  "similar_story": "<similar_story_text_1>",
  "dissimilar_story": "<dissimilar_story_text_1>"
 },
 {
  "anchor_story": "<anchor_story_text_2>",
  "similar_story": "<similar_story_text_2>",
  "dissimilar_story": "<dissimilar_story_text_2>"
 },
  ...
}
```

Document: {Source Document}

Table 5: Prompt Template for Synthetic Triple Generation