Efficient Paraphrase Detection With Embeddings

Giovanni Rocci Andrianos Michail Andreas Loizides Simon Clematide Juri Opitz
Department of Computational Linguistics
University of Zurich
<firstname.lastname>@uzh.ch

Abstract

Paraphrase detection is an important NLP task, related to plagiarism detection, paraphrase mining, and system evaluation. Most works train pairwise classifiers or prompts LLMs as classifiers. We shed the spotlight on a lightweight alternative: pre-trained embedding models combined with simple supervised calibration strategies. On PARAPHRASUS, a recent comprehensive paraphrase detection benchmark, our embedding-based methodology achieves accuracy levels competitive with published classifier and LLM baselines. Since embeddings can be computed once per text and reused for pair scoring, the approach is especially attractive for large-scale paraphrase classification and paraphrase mining. While our classification strategy itself is conceptually not novel, our work underpins its enduring value, by proving strong accuracy levels on par with costly recent LLM-based classifiers. We share our code and data under public license: https://github.com/impresso/paraphrasus.

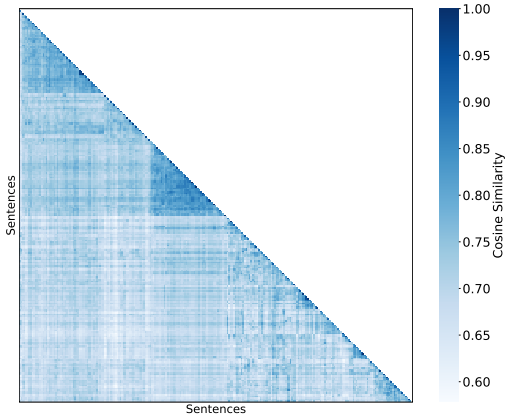

Figure 1: Cosine similarity heatmap for all unique combination of 250 individual STSB (Cer et al., 2017) sentences encoded with multilingual-e5-large-instruct (Wang et al., 2024). Rows and columns are ordered by embedding-based clustering; visible blocks indicate neighborhoods of mutually similar sentences and may partly reflect STSB's different source domains. The diagonal is excluded. A threshold of 0.95 would convert the continuous scores into binary labels.

1 Introduction

Paraphrase detection is the task of deciding whether two texts express equivalent meanings. The problem has a long history (c.f. Quintilianus, 95, p. 117) and remains central to NLP tasks such as plagiarism detection (where related passages must be assessed for their paraphrasticity; Barrón-Cedeño et al., 2013) and translation or summarization evaluation (where system outputs should be paraphrases of a reference; Zhou et al., 2006; Freitag et al., 2020).

Recent work, however, shows that paraphrase detection remains difficult: LLMs and especially trained classifiers can struggle to generalize across datasets (Michail et al., 2025). Many approaches also use pairwise cross-encoder classification, which is costly for large-scale use. Consider that in an all-pairs scenario where in a corpus all potential paraphrases must be found (Figure 1), a corpus of n texts yields $\mathcal{O}(n^2)$ potential candidate pairs. In this scenario, a cross-encoder classifier or prompted LLM must process each pair separately. Instead following the motivation of Sentence-BERT (Reimers and Gurevych, 2019), we note that an embedding-based approach computes one representation per text and can then reuse these representations for pair scoring with simple vector arithmetics (e.g., dot-produce, or cosine similarity), ultimately reducing model calls to $\mathcal{O}(n)$.

In our work, we investigate whether this advantage ports to our day and age — can we reliably and efficiently classify paraphrases with a simple embedding-based strategy, compared to more costly ones? To this aim, we evaluate recent multilingual embedding models for paraphrase detection and compare simple methods for mapping embedding pairs into binary decisions. Our results show that lightweight embedding-based methods can be

competitive with substantially more costly pairwise classifiers, especially in large-scale settings where reusable embeddings offer a clear practical advantage.

2 Related Work

Paraphrase detection supports a wide range of NLP applications, including semantic search (Reimers and Gurevych, 2019), style transfer (Krishna et al., 2020), machine translation and paraphrase generation (Madnani et al., 2007; Mallinson et al., 2017), plagiarism detection (Barrón-Cedeño et al., 2013; Sharjeel et al., 2016), text reuse (Büchler et al., 2012), and NLG evaluation (Freitag et al., 2020; Thompson and Post, 2020; Nawrath et al., 2024).

However, "predicting paraphrases is not easy" (Vahtola et al., 2022), and even strong classifiers and LLMs can struggle to generalize (Michail et al., 2025). Efficiency is another bottleneck, especially for pairwise classification at scale (Reimers and Gurevych, 2019) and mining scenarios. Multi-stage retrieval-and-reranking approaches address this by balancing accuracy and efficiency (Novais et al., 2026; Sasazawa et al., 2023): an initial embedding-based retrieval step narrows the candidate pairs, which a more computationally costly encoder then reranks. Other work optimizes this further into a single-step, end-to-end process; Bhat et al. (2025), for instance, use a unified encoder-decoder architecture in which encoder representations support retrieval and decoder representations support reranking, though at the cost of training such a complex architecture. Our approach is likewise single-stage, but requires only minimal calibration and no fine-tuning: rather than reranking a shortlist, we treat the calibrated embedding score itself as the final decision, maximizing efficiency at the potential cost of the accuracy a reranking stage would otherwise recover.

3 Method

We perform paraphrase detection with pre-trained embedding models and adapt them to binary classification using two lightweight supervised calibration strategies.

Calibration. Given embeddings for a candidate sentence pair, we first L2-normalize each embedding. We then map them to a binary paraphrase decision in two ways. First, we estimate a single threshold on the cosine similarity between the two embeddings. Second, we fit a logistic regression classifier on the unsigned element-wise difference between the embeddings. We motivate this choice empirically: in a controlled comparison (Appendix A), the unsigned difference achieves lower error on PARAPHRASUS than the signed difference, concatenation, or sum of the two embeddings. The first calibration method imposes a simple monotonic decision rule, while the second allows a more flexible decision boundary.

Efficiency and Scalability. Standard pairwise classifiers and prompted LLMs process each candidate pair separately. This leads to $\mathcal{O}(n^2)$ model calls when comparing all possible pairs in a corpus of n texts. In contrast, our method computes one embedding per text and reuses these embeddings for pair scoring. This reduces the number of model calls to $\mathcal{O}(n)$; the remaining pairwise comparisons are cheap vector operations.

4 Experimental Setup

Benchmark. We evaluate on PARAPHRASUS (Michail et al., 2025), a paraphrase detection benchmark with 10 datasets: 8 repurposed datasets and 2 newly annotated ones. An overview of the datasets is available in Table 1. To adapt semantic similarity datasets to paraphrase detection, Michail et al. (2025) relabeled the high-similarity sentence pairs as either paraphrases or non-paraphrases, while the low-similarity ones were considered non-paraphrases. All sentence pairs from NLI datasets were considered non-paraphrases instead. The datasets are organized into three challenge groups: "Classify" for binary paraphrase classification, "Minimize positive predictions" for avoiding false positives on non-paraphrases, and "Maximize positive predictions" for avoiding false negatives on true paraphrase pairs. For example, the *Maximize* group includes TRUE, whose positive pairs are derived from *Abstract Meaning Representation* guidelines (Banarescu et al., 2013).

The *Maximize* and *Minimize* groups test asymmetric failure modes that balanced classification can hide. *Minimize* is relevant for high-precision settings such as plagiarism detection, where false positives are costly. *Maximize* is relevant for paraphrase mining and semantic search, where false negatives reduce recall. Together, these settings test whether a method remains robust under skewed label distributions, where precision or recall may matter more depending on the application context.

Group	Dataset	Source Task	Size
Clfy	PAWS-X	Paraphrase Detection	14,000
	MRPC	Paraphrase Detection	1,725
	STS-H	Semantic Similarity	338
Min	SNLI	Natural Language Inference	6,632
	ANLI	Natural Language Inference	798
	XNLI	Natural Language Inference	16,700
	STS	Semantic Similarity	706
	SICK	Semantic Similarity	2,305
Max	TRUE*	Paraphrase Detection	167
	SIMP*	Readability Level	600

Table 1: Original purpose and size of datasets from the PARAPHRASUS benchmark. A $\star$ indicates newly annotated datasets.

Model	Size	Dim.	Max tok.	Pooling
M-MiniLM	118M	384	128	Mean
M-MPNet	278M	768	128	Mean
mGTE	305M	768	8192	[CLS] tkn.
KaLM-Emb	494M	896	512	Mean
mE5/mE5-instr	560M	1024	512	Mean
BGE-m3	570M	1024	8192	[CLS] tkn.
Jina-v3	570M	1024	8192	Mean
Qwen3-Emb	596M	1024	32K	[EOS] tkn.

Table 2: Specifications of evaluated embedding models.

Calibration and Evaluation. We use a leave-one-dataset-out procedure. For each PARAPHRASUS dataset, we hold out that dataset for testing and calibrate the threshold or logistic regression model on up to 500 samples from each of the remaining nine datasets. This cap avoids biasing calibration toward larger datasets and is close to the size of the smallest dataset.

Following Michail et al. (2025), we report percentage error rates. We first average dataset-level error rates within each challenge group and then compute the global score as the mean of the three group averages. Lower scores are better.

Baselines and Embedding Models. We compare against the models reported by Michail et al. (2025). These include a multilingual XLM-RoBERTa sequence classifier fine-tuned on PAWS-X and Llama3 Instruct 8B used as a prompted classifier. The Llama3 baseline was tested with three prompt formulations, asking whether the two sentences are *"a paraphrase"* (Prompt 1), *"semantically equivalent"* (Prompt 2) or *"about the same content"* (Prompt 3), with and without few-shot examples from PAWS-X, described in Michail et al. (2025).

We select text embedding models that are open-source, multilingual, relatively small ($< 1B$ parameters), and strong on the Massive Text Embedding Benchmark (MTEB) (Muennighoff et al., 2023). The encoder models are Paraphrase-Multilingual-MPNET-Base-V2 (M-MPNet), Paraphrase-Multilingual-MiniLM (M-MiniLM) (Reimers and Gurevych, 2019, 2020), Multilingual-E5-large in base and instruct variants (mE5, mE5-instr) (Wang et al., 2024), BGE-m3 (Chen et al., 2024), gte-multilingual-base (mGTE) (Zhang et al., 2024), and Jina Embeddings v3 (Jina-v3) (Sturua et al., 2024). We also evaluate two decoder-only embedding models derived from LLMs: Qwen3-Embedding-0.6B, based on Qwen3 (Zhang et al., 2025), and KaLM-Embedding-Multilingual-Mini-Instruct-V2.5, based on Qwen2 (Zhao et al., 2025). Table 2 summarizes their main specifications.

5 Results

Table 3 shows the main results. Calibrated embedding methods achieve overall error rates between 19.8 and 28.6. Threshold calibration performs better than logistic-regression calibration on average, with 24.0 versus 26.4 overall error. The best embedding configuration is mGTE with threshold calibration, which reaches an overall error of 19.8 and is slightly better than the strongest published baseline in this table, Llama3 (P1), at 20.9.

Performance varies across challenge groups. In *Classify*, embedding methods remain much higher XLM-R, which was fine-tuned on PAWS-X, but are closer to the Llama3 baselines. In *Minimize*, logistic-regression-calibrated embeddings obtain very low error rates on average, even beating Llama3 (P2) and Llama3 (P3), albeit by a small margin. In *Maximize*, threshold calibration is much stronger than logistic-regression calibration and several models come close to Llama3 (P1) on average. The best individual *Maximize* score is obtained by mE5-instr with threshold calibration.

6 Discussion

Calibration Method Effectiveness. Threshold calibration performs better than Logistic Regression calibration overall. This is mainly due to the *Maximize* group, where LR calibration has much higher error rates. By contrast, LR calibration performs well in the *Minimize* and *Classify* groups, being slightly better than threshold calibration in the latter.

		Classify!			Minimize!					Maximize!		Averages			
	Model	PAWSX	MRPC	STS-H	SNLI	ANLI	XNLI	STS	SICK	TRUE	SIMP	Clfy	Min	Max	$\overline{Err}$
Baselines	XLM-R← *PAWS*	**15.2**	54.1	33.4	32.4	7.2	26.7	46.6	37.0	31.4	5.3	**34.2**	30.0	18.3	27.5
	Llama3 (P1)	44.7	56.2	**23.6**	7.3	13.0	12.3	12.9	0.9	**9.0**	14.7	41.5	9.3	11.8	20.9
	Llama3 (P2)	40.7	37.6	45.9	**1.0**	1.2	1.4	2.4	0.1	34.7	47.3	41.4	1.2	41.0	27.9
	Llama3 (P3)	38.1	41.7	37.5	1.3	1.7	1.3	3.5	**0.0**	35.3	37.5	39.1	1.6	36.4	25.7
Threshold	M-MiniLM	52.4	38.3	51.8	6.3	2.0	2.2	4.2	3.1	33.5	8.5	47.5	3.6	21.0	24.0
	M-MPNet	51.8	**37.2**	54.4	6.2	2.0	2.3	4.1	3.6	29.9	8.3	47.8	3.6	19.1	23.5
	mGTE	49.4	39.2	52.7	2.3	0.9	0.5	1.0	1.6	16.2	5.8	47.1	1.3	11.0	**19.8**⋆
	KaLM-Emb	46.8	49.6	52.4	2.8	0.9	1.2	3.3	3.2	47.3	20.7	49.6	2.3	34.0	28.6
	mE5	52.5	39.7	55.6	4.6	1.6	2.1	3.4	1.9	41.3	**3.2**	49.3	2.7	22.2	24.7
	mE5-instr	53.7	43.2	52.7	4.4	1.4	2.0	5.0	2.5	9.6	10.5	49.9	3.1	**10.1**	21.0
	BGE-m3	50.5	51.0	51.8	2.9	0.8	0.7	4.7	2.8	22.2	18.2	51.1	2.4	20.2	24.6
	Jina-v3	53.1	42.1	55.3	4.2	2.5	3.4	2.1	2.3	26.3	5.0	50.2	2.9	15.7	22.9
	Qwen3-Emb	49.1	48.2	43.8	2.4	2.1	1.6	1.7	3.4	47.9	16.8	47.0	2.2	32.4	27.2
	Threshold Calibration Averages:											48.8	2.7	20.6	24.0
Logistic Regression	M-MiniLM	52.1	48.2	39.3	2.4	1.1	0.5	1.3	1.4	48.5	27.5	46.5	1.3	38.0	28.6
	M-MPNet	52.1	47.8	45.0	2.2	0.9	0.4	0.7	1.2	46.7	25.2	48.3	1.1	36.0	28.5
	mGTE	49.5	47.9	42.6	1.7	0.5	**0.2**	0.4	1.2	34.1	17.8	46.7	0.8	26.0	24.5
	KaLM-Emb	46.9	49.4	46.4	1.9	0.5	0.8	2.5	1.6	42.5	23.0	47.6	1.5	32.8	27.3
	mE5	51.2	48.3	40.5	1.7	0.4	0.3	0.4	0.7	53.3	18.8	46.7	0.7	36.0	27.8
	mE5-instr	53.0	49.4	35.8	1.2	**0.3**	0.3	0.7	0.8	32.3	19.8	46.1	0.7	26.0	24.3
	BGE-m3	50.2	50.1	39.3	1.9	0.5	0.4	0.7	1.4	30.5	17.5	46.5	1.0	24.0	23.8
	Jina-v3	52.2	45.8	40.8	1.3	0.4	0.5	**0.3**	0.1	38.3	20.7	46.3	**0.5**	29.5	25.4
	Qwen3-Emb	49.7	48.3	38.2	1.6	1.0	1.0	0.8	2.1	53.3	19.5	45.4	1.3	36.4	27.7
	Logistic Regression Calibration Averages:											46.7	1.0	31.6	26.4

Table 3: Main results on PARAPHRASUS of various models. Lower numbers are better.

One possible explanation is that the threshold rule acts as a strong regularizer. It uses a single monotonic decision boundary over cosine similarity and may therefore transfer better across heterogeneous datasets. LR calibration is more flexible because it assigns weights to individual embedding dimensions, but this flexibility may also make it more sensitive to dataset-specific calibration artifacts. This could explain why it performs well for some objectives but generalizes poorly in the *Maximize* group.

Cross-Lingual Matching. PARAPHRASUS includes multilingual datasets such as PAWS-X and XNLI, but its sentence pairs are language-internal. We therefore test whether calibration on monolingual pairs transfers to cross-lingual paraphrase pairs. For this stress test, we use TaPaCo (Scherrer, 2020), which contains clusters of equivalent sentences across 73 languages. We construct cross-lingual positive pairs, calibrate on all PARAPHRASUS datasets, and treat TaPaCo as a fully held-out evaluation set. Results are shown in Table 4.

Overall performance varies considerably across models. Qwen3-Emb reaches 83.5% error with threshold calibration and 91.2% with LR calibration, while BGE-m3 achieves the lowest threshold error at 36.9%. Model size does not predict cross-lingual transfer: BGE-m3 (570M) and Qwen3-Emb (596M) are comparable in size but lie at opposite ends of the performance range. These results suggest that monolingual calibration does not automatically transfer to cross-lingual paraphrase pairs, and that training objective or multilingual alignment may matter more than parameter count.

Model	Threshold (%)	LR Classifier (%)
M-MiniLM	37.6	53.8
M-MPNet	37.3	**51.2**
mGTE	48.1	78.1
KaLM-Emb	77.5	85.5
mE5	76.7	85.4
mE5-instr	53.1	81.8
BGE-m3	**36.9**	66.7
Jina-v3	53.5	72.6
Qwen3-Emb	83.5	91.2

Table 4: Error rate (%) for models on TaPaCo dataset.

Embedding Prompt Optimization. The multilingual E5-instruct embedding model supports task-specific prompts that influence the resulting embeddings. In the main evaluation, we use the default empty prompt configured by the model authors. We additionally test the three paraphrase-oriented prompts used for Llama3 by Michail et al. (2025),

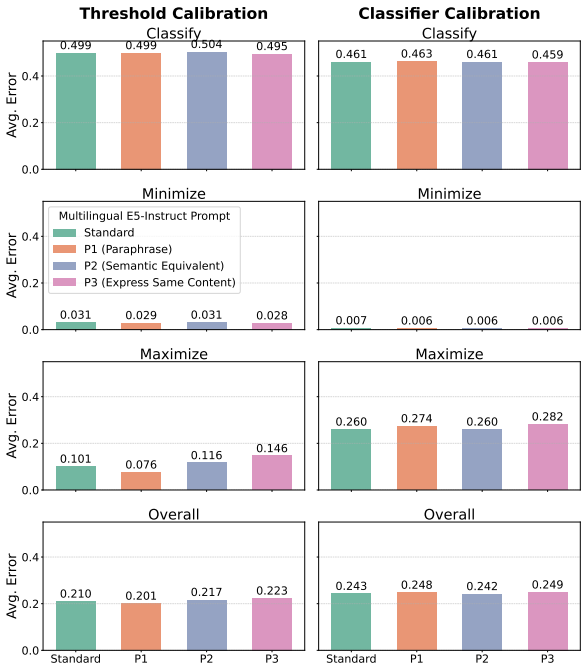

Figure 2: Average error rates for mE5-instr prompts.

which express different notions of paraphrase strictness. Figure 2 shows the results by PARAPHRASUS objective and calibration method.

Prompt effects depend on the calibration method. With threshold calibration, Prompt 1, asking whether the sentences are "a paraphrase", has the lowest overall error among the custom prompts. Prompt 3, asking whether they are "about the same content", is best for *Classify*. For logistic regression calibration instead, Prompt 2, asking whether they are "semantically equivalent", is tied with the standard prompt for *Maximize* and best overall. The strength of the direct paraphrase prompt is consistent with the Llama3 results reported by Michail et al. (2025).

Calibration Sample Efficiency. As explained in Section 4, we sample 500 labeled sentence pairs from all datasets but one for calibration. To assess whether similar results can be obtained with a different amount of supervision, we repeat calibration with 25, 50, 100, 250, 500, 750, and 1000 samples per dataset (capped at the dataset's available size for datasets smaller than the requested amount), using three representative models: the best-performing model, mGTE; the smallest, multilingual-MiniLM; and the largest, Qwen3-Embedding. Figure 3 shows average error rate for both calibration methods across sample sizes.

Threshold calibration seems to be robust to calibration set size: error rates fluctuate only mildly

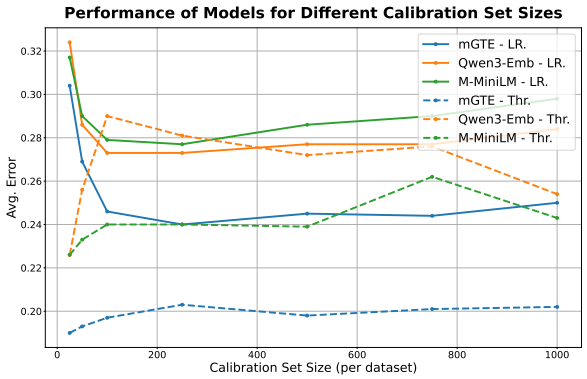

Figure 3: Average error rate for three models for both calibration methods using different sample sizes.

across the full range of sample sizes tested, and interestingly reach minimum values already at 25 samples per dataset. In the case of Qwen3-Embeddings there is a sharp performance degradation when increasing from 25 to 100 samples, but it stabilizes for bigger sample sizes. Logistic regression calibration is more sensitive: error rates are substantially higher at 25–50 samples for all three models, drop sharply by 100 samples, and continue to improve up to 250 samples before ticking slightly upward at 500-750 and increasing more sharply towards 1000 samples, a pattern that may indicate diminishing returns from additional calibration data due to overfitting and shift of the data distribution in favor of larger datasets, though we do not test this directly. Overall, these results suggest that far fewer labeled pairs than our default 500 per dataset are sufficient for threshold calibration, while logistic regression benefits from a larger calibration set up to a point.

7 Conclusions

We evaluated pre-trained multilingual embedding models for paraphrase detection using two lightweight supervised calibration strategies. The results show that calibrated embeddings can be competitive with published classifier and LLM baselines on PARAPHRASUS, while offering a clear efficiency advantage for large-scale all-pairs settings. In particular, threshold calibration is simple, effective, and often more robust than logistic-regression calibration.

Limitations

The studied approach is not fully unsupervised: both threshold and logistic-regression calibration require labeled data. Although a calibrated thresh-

old can be reused for new data, its reliability may decrease under domain or language shift, as suggested by our cross-lingual stress test. Future work should therefore explore unsupervised or minimally supervised threshold selection.

A second limitation is that embedding-based classification itself is not a new paradigm. Our contribution is instead an empirical evaluation of this paradigm for paraphrase detection, where pairwise classifiers and LLMs are more commonly used. Finally, we evaluate only models under 1B parameters. Since embedding models and LLMs continue to develop rapidly, ongoing evaluation will be needed to track the performance-efficiency tradeoff across newer systems.

A third limitation relates to interpretability and error analysis, as interpretability methods that are applicable to classifiers do not straightforwardly transfer to tracing decisions through embedding models. However, work may draw methods or intuition from recent research on embedding and similarity interpretability (Opitz et al., 2025, 2026).

Acknowledgements

This work received funding through the project *Impresso – Media Monitoring of the Past II Beyond Borders: Connecting Historical Newspapers and Radio*. Impresso is a research project funded by the Swiss National Science Foundation (SNSF 213585) and the Luxembourg National Research Fund (17498891).

References

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Alberto Barrón-Cedeño, Marta Vila, M. Antònia Martí, and Paolo Rosso. 2013. Plagiarism meets paraphrasing: Insights for the next generation in automatic plagiarism detection. *Computational Linguistics*, 39(4):917–947.

Riyaz Ahmad Bhat, Jaydeep Sen, Rudra Murthy, and Vignesh P. 2025. UR2N: Unified retriever and ReraNker. In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pages 595–602, Abu Dhabi, UAE. Association for Computational Linguistics.

Marco Büchler, Gregory Crane, Maria Moritz, and Alison Babeu. 2012. Increasing recall for text re-use in historical documents to support research in the humanities. In *Theory and Practice of Digital Libraries: Second International Conference, TPDL 2012, Paphos, Cyprus, September 23-27, 2012. Proceedings 2*, pages 95–100. Springer.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-embedding: Multi-Linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2318–2335, Bangkok, Thailand. Association for Computational Linguistics.

Markus Freitag, George Foster, David Grangier, and Colin Cherry. 2020. Human-paraphrased references improve neural machine translation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1183–1192, Online. Association for Computational Linguistics.

Kalpesh Krishna, John Wieting, and Mohit Iyyer. 2020. Reformulating unsupervised style transfer as paraphrase generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 737–762, Online. Association for Computational Linguistics.

Nitin Madnani, Necip Fazil Ayan, Philip Resnik, and Bonnie Dorr. 2007. Using paraphrases for parameter tuning in statistical machine translation. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 120–127, Prague, Czech Republic. Association for Computational Linguistics.

Jonathan Mallinson, Rico Sennrich, and Mirella Lapata. 2017. Paraphrasing revisited with neural machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 881–893, Valencia, Spain. Association for Computational Linguistics.

Andrianos Michail, Simon Clematide, and Juri Opitz. 2025. PARAPHRASUS: A comprehensive benchmark for evaluating paraphrase detection models. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 8749–8762, Abu Dhabi, UAE. Association for Computational Linguistics.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. MTEB: Massive text embedding benchmark. In *Proceedings of the 17th Conference*

of the European Chapter of the Association for Computational Linguistics, pages 2014–2037, Dubrovnik, Croatia. Association for Computational Linguistics.

Marcel Nawrath, Agnieszka Nowak, Tristan Ratz, Danilo Walenta, Juri Opitz, Leonardo Ribeiro, João Sedoc, Daniel Deutsch, Simon Mille, Yixin Liu, Sebastian Gehrmann, Lining Zhang, Saad Mahamood, Miruna Clinciu, Khyathi Chandu, and Yufang Hou. 2024. On the role of summary content units in text summarization evaluation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 272–281, Mexico City, Mexico. Association for Computational Linguistics.

Artur M. A. Novais, Anna P. V. L. B. Moreira, Maria C. X. de Almeida, João P. C. Presa, Fernando M. Federson, and Sávio S. T. de Oliveira. 2026. Optimizing efficiency in multi-stage semantic re-ranking architectures. In *Proceedings of the 17th International Conference on Computational Processing of Portuguese (PROPOR 2026) - Vol. 1*, pages 562–571, Salvador, Brazil. Association for Computational Linguistics.

Juri Opitz, Andrianos Michail, Lucas Moeller, Sebastian Padó, and Simon Clematide. 2026. Similar, but why? a toolkit for explaining text similarity. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 203–214, Rabat, Marocco. Association for Computational Linguistics.

Juri Opitz, Lucas Moeller, Andrianos Michail, Sebastian Padó, and Simon Clematide. 2025. Interpretable text embeddings and text similarity explanation: A survey. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 22303–22319, Suzhou, China. Association for Computational Linguistics.

Marcus Fabius Quintilianus. 95. *Institutio Oratoria (Book X)*, loeb classical library edition, 1920 edition. Loeb. Public domain.

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

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.

Yuichi Sasazawa, Kenichi Yokote, Osamu Imaichi, and Yasuhiro Sogawa. 2023. Text retrieval with multi-stage re-ranking models. *Preprint*, arXiv:2311.07994.

Yves Scherrer. 2020. TaPaCo: A corpus of sentential paraphrases for 73 languages. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 6868–6873, Marseille, France. European Language Resources Association.

Muhammad Sharjeel, Paul Rayson, and Rao Muhammad Adeel Nawab. 2016. UPPC - Urdu paraphrase plagiarism corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 1832–1836, Portorož, Slovenia. European Language Resources Association (ELRA).

Saba Sturua, Isabelle Mohr, Mohammad Kalim Akram, Michael Günther, Bo Wang, Markus Krimmel, Feng Wang, Georgios Mastrapas, Andreas Koukounas, Nan Wang, and Han Xiao. 2024. jina-embeddings-v3: Multilingual embeddings with task LoRA. *Preprint*, arXiv:2409.10173.

Brian Thompson and Matt Post. 2020. Automatic machine translation evaluation in many languages via zero-shot paraphrasing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 90–121, Online. Association for Computational Linguistics.

Teemu Vahtola, Mathias Creutz, and Jörg Tiedemann. 2022. It is not easy to detect paraphrases: Analysing semantic similarity with antonyms and negation using the new SemAntoNeg benchmark. In *Proceedings of the Fifth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 249–262, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Multilingual e5 text embeddings: A technical report. *Preprint*, arXiv:2402.05672.

Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, Meishan Zhang, Wenjie Li, and Min Zhang. 2024. mGTE: Generalized long-context text representation and reranking models for multilingual text retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1393–1412, Miami, Florida, US. 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. *arXiv preprint*. ArXiv:2506.05176 [cs].

Xinping Zhao, Xinshuo Hu, Zifei Shan, Shouzheng Huang, Yao Zhou, Xin Zhang, Zetian Sun, Zhenyu Liu, Dongfang Li, Xinyuan Wei, Youcheng Pan,

Yang Xiang, Meishan Zhang, Haofen Wang, Jun Yu, Baotian Hu, and Min Zhang. 2025. KaLM-Embedding-V2: Superior Training Techniques and Data Inspire A Versatile Embedding Model. *arXiv preprint*. ArXiv:2506.20923 [cs].

Liang Zhou, Chin-Yew Lin, Dragos Stefan Munteanu, and Eduard Hovy. 2006. ParaEval: Using paraphrases to evaluate summaries automatically. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 447–454, New York City, USA. Association for Computational Linguistics.

A Alternate Logistic Regression Design Variants

Used Feature	Model	$\overline{Clfy}$	$\overline{Min}$	$\overline{Max}$	$\overline{Err}$
Concatenation	M-MiniLM	47.9	19.1	95.5	54.2
	M-MPNET	48.5	20.3	95.2	54.7
	mGTE	49.0	22.1	95.8	55.6
	KaLM-Emb	45.6	26.5	99.4	57.2
	mE5	47.5	22.9	99.8	56.7
	mE5-instr	47.3	21.7	99.9	56.3
	BGE-m3	46.6	20.5	97.8	55.0
	Jina-v3	45.5	19.8	95.5	53.6
	Qwen3-Emb	48.2	20.0	96.3	54.8
Signed Diff.	M-MiniLM	48.1	4.4	100.0	50.8
	M-MPNET	47.9	4.1	100.0	50.7
	mGTE	47.9	2.4	100.0	50.1
	KaLM-Emb	47.9	4.0	100.0	50.6
	mE5	47.9	0.7	100.0	49.5
	mE5-instr	47.9	2.9	100.0	50.3
	BGE-m3	47.9	4.2	100.0	50.7
	Jina-v3	48.0	4.2	100.0	50.7
	Qwen3-Emb	47.9	4.2	100.0	50.7
Sum	M-MiniLM	48.1	19.1	95.5	54.2
	M-MPNET	48.6	20.6	93.2	54.1
	mGTE	49.4	22.9	94.7	55.7
	KaLM-Emb	45.2	27.4	99.1	57.2
	mE5	47.5	23.7	99.5	56.9
	mE5-instr	47.1	22.4	99.8	56.4
	BGE-m3	46.5	21.4	97.1	55.0
	Jina-v3	45.4	20.3	95.0	53.6
	Qwen3-Emb	48.2	20.7	94.4	54.4

Table 5: Mean error rates for the evaluated models on the three PARAPHRASUS tasks, using logistic regression calibration fitted on the concatenation, on the signed difference and on the sum of the embeddings of a sentence pair.