ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 979-8-3315-6701-9/26/$31.00 ©2026 IEEE | DOI: 10.1109/ICASSP55912.2026.11465138

LOGIC-ORIENTED RETRIEVER ENHANCEMENT VIA CONTRASTIVE LEARNING

Wenxuan Zhang[1], *Yuan-Hao Jiang*[1], *Changyong Qi*[1], *Rui Jia*[1], *Yonghe Wu*[2]

[1]Shanghai Institute of Artificial Intelligence for Education, East China Normal University, China
[2]Education Technology, East China Normal University, China

ABSTRACT

Large language models (LLMs) struggle in knowledge-intensive tasks, as retrievers often overfit to surface similarity and fail on queries involving complex logical relations. The capacity for logical analysis is inherent in model representations but remains underutilized in standard training. LORE (Logic ORiented Retriever Enhancement) introduces fine-grained contrastive learning to activate this latent capacity, guiding embeddings toward evidence aligned with logical structure rather than shallow similarity. LORE requires no external supervision, resources, or pre-retrieval analysis, remains index-compatible, and consistently improves retrieval utility and downstream generation while maintaining efficiency. The datasets and code are publicly available at `https://github.com/mazehart/Lore-RAG`.

***Index Terms*—** Large Language Models (LLMs), Retrieval-Augmented Generation (RAG), Contrastive Learning

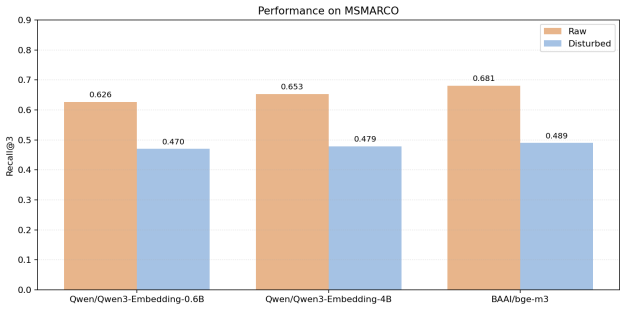

Fig. 1. **Impact of complex logical expressions on embedding model performance.** Results on MSMARCO dataset show significant performance degradation across different embedding models when queries are disturbed with complex logical expressions, highlighting the vulnerability of current embedding approaches to complex logical structures.

1. INTRODUCTION

Large language models (LLMs) have achieved significant progress across a wide range of natural language processing tasks [1, 2], yet they remain limited in knowledge-intensive scenarios, often exhibiting factual hallucinations, stale knowledge, and poor traceability. These limitations reduce reliability and make it difficult to deploy LLMs in high-stakes applications where correctness and verifiability are critical [3, 4]. Therefore, retrieval-augmented generation (RAG) has emerged as a practical remedy [5]. By following a "retrieve-then-generate" paradigm, RAG retrieves external knowledge sources through embedding models via vector matching and then passes them to large models, thereby mitigating hallucination problems and enhancing interpretability and source traceability [6, 7].

In practice, a persistent gap exists: while embedding models perform well on direct and straightforward queries, their effectiveness significantly degrades when handling queries disturbed with complex logical expressions. As shown in the figure, existing open-source embedding models, even with large parameter counts, cannot avoid the difficulties brought by queries disturbed with complex logical expressions. This is because their similarity-driven nature causes them to often favor surface-level lexical overlap rather than faithfully representing and satisfying the complex logical requirements of queries, leading to fragile evidence sets that fail to meet query requirements [8].

A natural response is to analyze or decompose queries using powerful LLMs before retrieval [9, 10]. However, this "LLM-in-the-loop" strategy introduces non-trivial costs and latency, and relies on knowledge graphs and other structures constructed for datasets. This paper takes a different path: rather than adding more LLM computation, we pose a question—do existing open-source embedding models already contain untapped capacity to represent complex linguistic structures? Can targeted contrastive learning activate this capacity, enabling retrievers to directly encode complex logical structures and salience rather than relying solely on surface similarity?

This paper presents LORE, an embedding enhancement method through contrastive learning for improving retrievers. Based on existing datasets, LORE does not invoke large models to assist embedding after queries and before retrieval, but rather retrains the embedding model using fine-grained contrastive knowledge annotated by large models after pretraining, emphasizing signals related to complex logical expres-

Corresponding author: yhwu@deit.ecnu.edu.cn

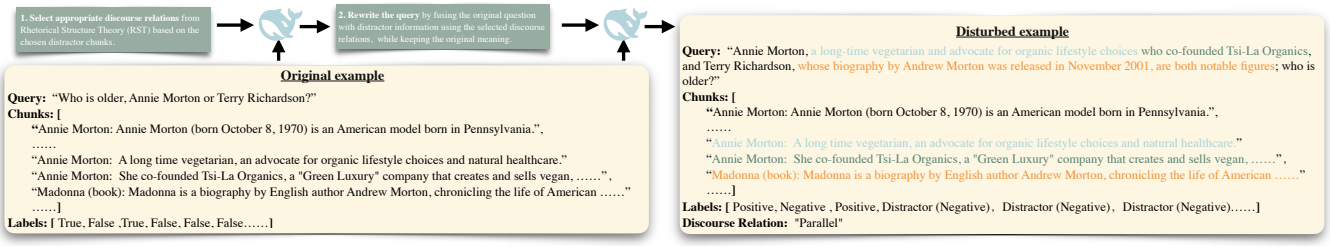

Fig. 2. An illustration of **Query Rewriting**. The left side shows an original example with query, chunks, and labels, while the right side presents the corresponding modified example with enriched query, discourse relation, and distractor annotations.

sions.

The contributions of this paper are three-fold:

- This paper proposes LORE, an embedding model enhancement paradigm that integrates large model knowledge in the post-pretrain stage, providing a simple contrastive fine-tuning method.
- This paper constructs an open-source fine-grained contrastive learning dataset based on existing open-source.
- This paper validates the consistent gains of LORE on multiple tasks featuring queries disturbed with complex logical expressions, with improvements on unlabeled tasks particularly demonstrating the effectiveness of the proposed method.

2. RELATED WORK

Beyond similarity. Recent methods have begun to address the limitations caused by RAG's reliance on similarity. METRAG attempts to use changes in LLM answer accuracy on downstream tasks as supervision signals to fine-tune the retrieval embedding model, training it to identify retrieval content that helps improve LLM answer accuracy on downstream tasks [10]. Meanwhile, Mindful-RAG combines LLM's internal knowledge with external knowledge graphs, first analyzing query intent through LLMs, then integrating knowledge graphs to introduce context alignment and result verification in the retrieval process, thereby reducing errors and ambiguity in complex reasoning [9].

Limitations. These methods improve utility through task feedback, intent modeling, or external knowledge structures, but often introduce additional overhead and rely on external supervision or resources, hindering transferability across different tasks. While they leverage large model knowledge to enhance embedding model performance, the fundamental capabilities of embedding models are not improved in this process. Therefore, a key question is: how to equip retrievers with understanding of queries with complex logical expressions, enabling them to directly encode queries and prioritize evidence that conforms to the query's logical structure, while maintaining efficiency and avoiding dependence on pre-retrieval LLM analysis, external supervision, and external knowledge structures [8].

Positioning. This paper proposes LORE, a contrastive post-training method to activate the latent capacity of open-source embedding models in representing queries with complex logical expressions, enabling retrievers to directly encode logical structure rather than relying on surface similarity, thereby improving retrieval performance at the fundamental capability level of embeddings.

3. METHOD

3.1. Dataset Construction

Fine-grained Chunk Categories The construction starts from a base corpus with query q, candidate chunks $\{c_k\}$, and boolean labels $y_k \in \{\text{True}, \text{False}\}$. For each q, a subset of False-labeled chunks is sampled as distractors, each selected distractor is assigned its original chunk index as a source identifier, and an LLM is prompted to rewrite q into a more natural query that embeds some distractor content while preserving the original meaning. The distractor usage is then mapped back to chunk indices and three tiers are constructed for each q and candidate set $\mathcal{C}(q)$:

- *Positive (P)*: chunks with $y_k = \text{True}$ that are sufficient to answer the query.

- *Distractor (N1)*: False-labeled chunks that were used by the LLM in the rewritten question as distractors, appearing relevant to the question but providing no help in answering it.

- *Negative (N2)*: other False-labeled chunks not used in the rewrite.

The three tiers are denoted as P, N1, and N2. When only a portion of the dataset is rewritten, the remaining samples retain their original queries and boolean labels, which are mapped to these tiers for consistency.

Query Rewriting The dataset construction follows a two-step process as illustrated in Figure 2. First, appropriate discourse relations from Rhetorical Structure Theory (RST) [11] are selected based on the chosen chunks. Eight common relations—Sequential, Transitional, Supplementary, Contrastive, Causal, Parallel, Hypothetical, and Explanatory—serve as

guidelines. Second, the query is rewritten by fusing the original question with distractor information using the selected discourse relations, resulting in a modified example with enriched query and discourse relation annotations. The constructed training dataset, CoEnTrain, is released.

3.2. Contrastive Learning

Contrastive loss The framework employs two separate encoders: a query encoder $\mathcal{M}_q$ and a document encoder $\mathcal{M}_d$. Queries and chunks are embedded into a shared vector space using the respective encoders. Let $\mathbf{h}_q = \mathcal{M}_q(q)$ and $\mathbf{h}_k = \mathcal{M}_d(c_k)$ denote the normalized embeddings, and s_k the cosine similarity between the query and the k-th candidate chunk:

$$\mathbf{h}_q,\ \mathbf{h}_k \in \mathbb{R}^d,\ s_k = \cos(\mathbf{h}_q, \mathbf{h}_k).$$

For each $(q, \mathcal{C}(q))$, temperature scaling and negative-tier weighting are applied in logit space. Let $\tilde{s}_k$ denote the adjusted similarity:

$$\tilde{s}_k = \begin{cases} s_k/\tau, & k \in \mathrm{P} \\ s_k/\tau + \log(\beta), & k \in \mathrm{N1} \\ s_k/\tau + \log(\alpha), & k \in \mathrm{N2} \end{cases} \quad (\beta > \alpha > 0).$$

The differential weighting $\beta > \alpha$ is crucial for handling complex logical expressions, as it ensures that distractors (N1) receive stronger penalties than negatives (N2), forcing the model to learn fine-grained distinctions. The probability for a positive sample $k \in \mathrm{P}$ is computed using only negative samples (N1 and N2) in the denominator:

$$p_k = \frac{\exp(\tilde{s}_k)}{\sum_{t \in N1 \cup N2} \exp(\tilde{s}_t) + \exp(\tilde{s}_k)}, \quad k \in \mathrm{P}.$$

Following the InfoNCE framework [12], the objective minimized for a query q is

$$\mathcal{L}(q) = -\frac{1}{|P|} \sum_{k \in P} \log p_k.$$

This design scales distractor/negative samples in logit space (with $\beta > \alpha$), excludes positives from the denominator to avoid dilution, and emphasizes the hierarchical separation $\mathrm{P} \succ \mathrm{N1} \succ \mathrm{N2}$ to improve understanding of complex logical expressions.

Training process The document encoder $\mathcal{M}_d$ is kept frozen with pre-trained parameters, while the query encoder $\mathcal{M}_q$ is fine-tuned during training. Based on the embeddings and similarities defined above, the contrastive loss serves as the optimization objective. Formally, the query encoder parameters θ_q are optimized by

$$\theta_q^* = \arg\min_{\theta_q} \mathbb{E}_{q \sim \mathcal{D}}\, \mathcal{L}(q),$$

where $\mathcal{D}$ denotes the training set.

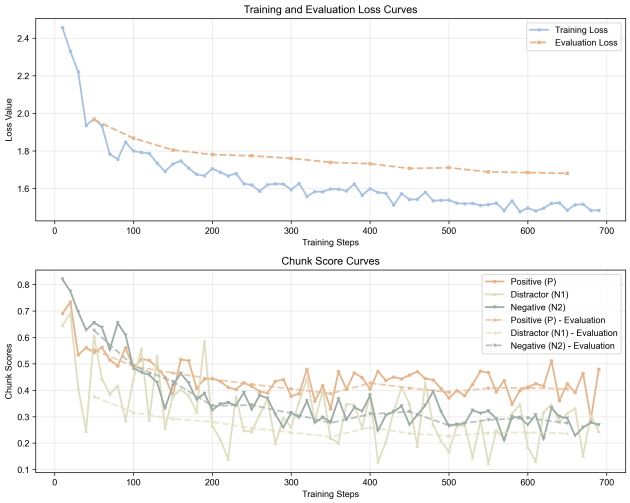

Fig. 3. Training and validation loss curves (upper) and chunk score evolution for different chunk categories (lower) during Qwen3-Embedding-0.6B fine-tuning.

4. EXPERIMENTS

4.1. Experimental Setup

Datasets Experiments are conducted on three widely-used public datasets: HotpotQA [13], MS MARCO [14], and MuSiQue [15]. For training and validation sets, Query Rewriting is performed and fine-grained category contrastive information is annotated on MS MARCO data, while HotpotQA and MuSiQue datasets remain unannotated. For test sets, Query Rewriting is applied to all datasets while preserving the original test sets for comparison. The rewritten test sets undergo manual screening to ensure they meet the rewriting requirements.

Models and Baselines Experiments are conducted using two embedding models: Qwen3 Embedding [16] and BGE M3-Embedding [17]. The effectiveness of LORE is compared against raw models and traditional contrastive learning approaches on both raw test sets and disturbed test sets.

Evaluation Metrics Retrieval performance is primarily evaluated using positive recall at top-k positions (@3, @5, @10), which measures the proportion of relevant chunks successfully retrieved within the top-k results. For Query Rewriting test sets, the recall rates of distractor samples are also recorded.

Experimental Setup Model training is conducted on a single A800 GPU. The learning rate is set to 1×10^{-5}, temperature parameter $\tau = 0.05$, and tier-weighted parameters $\alpha = 1.0$, $\beta = 3.0$. Training is performed for 1 epoch with batch size 32. For reproducibility, three models with different random seeds (0, 1, 2) are trained and the mean and standard deviation across runs are reported.

Table 1. Recall of positive chunks and distractor chunks.

Recall		HotpotQA			Musique			MSMARCO		
		Raw Dataset	Disturbed Dataset		Raw Dataset	Disturbed Dataset		Raw Dataset	Disturbed Dataset	
		P↑	P↑	N1↓	P↑	P↑	N1↓	P↑	P↑	N1↓
Qwen/Qwen3-Embedding-0.6B										
@3	**Raw Model**	56.37±0.00	33.76±0.00	70.03±0.00	59.07±0.00	39.55±0.00	83.88±0.00	62.63±0.00	46.98±0.00	84.25±0.00
	+InfoNCE	**70.87**±0.40	59.71±0.56	50.51±1.06	**67.23**±0.42	53.65±0.47	70.36±1.16	**68.39**±0.25	51.63±0.40	82.03±0.56
	+LORE	70.72±0.09	**65.45**±0.54	**23.81**±1.39	66.68±0.52	**59.28**±0.34	**36.87**±2.06	68.35±0.23	**69.30**±0.35	**20.15**±0.87
@5	**Raw Model**	68.04±0.00	49.23±0.00	81.59±0.00	70.13±0.00	56.38±0.00	88.80±0.00	82.41±0.00	73.25±0.00	93.34±0.00
	+InfoNCE	**81.96**±0.20	74.77±0.68	69.65±1.13	**79.93**±0.34	70.03±0.23	81.25±0.85	84.96±0.03	76.68±0.24	92.71±0.05
	+LORE	81.84±0.44	**77.79**±0.34	**39.73**±2.09	79.41±0.30	**73.32**±0.05	**52.81**±1.76	**85.06**±0.31	**85.77**±0.22	**44.22**±0.78
@10	**Raw Model**	80.55±0.00	69.13±0.00	90.96±0.00	84.28±0.00	77.46±0.00	92.09±0.00	96.75±0.00	96.70±0.00	99.45±0.00
	+InfoNCE	**92.61**±0.08	89.30±0.21	86.95±0.64	**92.81**±0.23	88.75±0.17	89.97±0.29	96.75±0.00	96.70±0.00	99.45±0.00
	+LORE	92.45±0.02	**89.62**±0.32	**62.84**±2.28	92.37±0.16	**88.87**±0.06	**74.66**±0.96	96.75±0.00	96.70±0.00	99.45±0.00
BAAI/bge-m3										
@3	**Raw Model**	64.68±0.00	40.04±0.00	71.84±0.00	60.12±0.00	42.33±0.00	84.07±0.00	68.10±0.00	48.95±0.00	84.02±0.00
	+InfoNCE	**71.53**±0.17	57.77±0.62	53.96±0.89	**66.51**±0.06	52.94±0.09	73.48±0.36	70.21±0.27	52.49±0.29	80.93±0.21
	+LORE	70.76±0.22	**65.57**±0.58	**20.43**±1.38	65.88±0.08	**59.94**±0.53	**25.80**±2.78	**70.52**±0.23	**69.28**±0.08	**22.80**±0.75
@5	**Raw Model**	76.09±0.00	58.24±0.00	82.89±0.00	71.96±0.00	60.28±0.00	88.23±0.00	85.25±0.00	75.20±0.00	93.51±0.00
	+InfoNCE	**83.06**±0.23	73.74±0.11	72.22±0.57	**77.85**±0.05	69.19±0.16	84.02±0.20	85.98±0.18	78.17±0.20	92.31±0.20
	+LORE	82.52±0.11	**77.59**±0.38	**33.05**±2.09	77.28±0.23	**72.29**±0.06	**40.00**±3.06	**86.20**±0.21	**86.10**±0.09	**44.89**±0.60
@10	**Raw Model**	88.28±0.00	78.86±0.00	92.47±0.00	86.58±0.00	81.27±0.00	91.69±0.00	96.75±0.00	96.70±0.00	99.45±0.00
	+InfoNCE	**93.26**±0.03	89.14±0.12	87.55±0.21	**91.69**±0.05	87.78±0.07	90.90±0.13	96.75±0.00	96.70±0.00	99.45±0.00
	+LORE	93.10±0.08	**89.49**±0.61	**54.90**±2.96	91.37±0.08	**88.36**±0.09	**61.53**±2.59	96.75±0.00	96.70±0.00	99.45±0.00

4.2. Results and Analysis

Training Dynamics Figure 3 shows the learning progression of LORE. Initially, the model favors distractor (N1) samples over positive (P) samples due to surface-level lexical overlap, despite N1 being irrelevant for answering queries.

During training, Distractor (N1) sample scores consistently decrease and eventually converge with negative (N2) samples at lower levels, while positive (P) samples maintain relatively stable scores and establish clear dominance. This progressive separation validates the effectiveness of the tier-weighted contrastive loss design.

Retrieval Performance As shown in Table 1, both InfoNCE and LORE significantly outperform the Raw Model on both Raw Dataset and Disturbed Dataset. Since only the query encoder $\mathcal{M}_q$ is fine-tuned in the experiments, this demonstrates that contrastive learning can enhance the model's understanding of queries, thereby improving retrieval performance. However, compared to InfoNCE, LORE, which utilizes fine-grained annotations from large language models, achieves more significant improvements.

Generalization Capability Only MSMARCO received fine-grained annotations in the training set, yet both HotpotQA and MuSiQue datasets also benefit from this approach. This indicates that the LORE method possesses cross-task generalization effects. When constructing sufficiently general and rich fine-grained datasets, broader performance improvements in embedding models across various tasks can be anticipated. This transferability demonstrates the practical value of LORE in real-world applications where comprehensive annotation of all target domains is often impractical.

5. LIMITATIONS AND CONCLUSION

Existing embedding models are limited by their training corpora and pretraining paradigms, resulting in insufficient comprehension of queries with complex logical expressions and an overreliance on surface-level similarity. Previous work has mainly focused on using large models in specific scenarios to enhance performance, rather than improving the foundational capabilities of embedding models. To address this, queries disturbed with complex logical expressions are constructed based on Rhetorical Structure Theory (RST), and fine-grained contrastive annotations across three chunk categories are applied. Contrastive learning enables embedding models to better capture complex logical expressions while remaining fully compatible with existing methods, reducing the need for additional task-specific designs.

Despite demonstrated effectiveness, the approach is constrained by the diversity of logical structures and datasets. Its generalization requires further investigation. Future directions include developing agent-based data annotation systems over more realistic queries with complex logical expressions, allowing the transfer of large models' query-intent understanding to smaller embedding models via contrastive learning, thereby strengthening their foundational capabilities.

6. ACKNOWLEDGMENTS

This work was supported by the Major Program of National Fund of Philosophy and Social Science of China (Grant No. 21&ZD238).

7. REFERENCES

[1] Tom B. Brown, Benjamin Mann, Nick Ryder, et al., "Language models are few-shot learners," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, Eds., 2020.

[2] Muhammad Arslan, Hussam Ghanem, Saba Munawar, and Christophe Cruz, "A survey on rag with llms," *Procedia computer science*, vol. 246, pp. 3781–3790, 2024.

[3] Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al., "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions," *ACM Transactions on Information Systems*, vol. 43, no. 2, pp. 1–55, 2025.

[4] Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen, "Halueval: A large-scale hallucination evaluation benchmark for large language models," in *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, Houda Bouamor, Juan Pino, and Kalika Bali, Eds. 2023, pp. 6449–6464, Association for Computational Linguistics.

[5] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al., "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in neural information processing systems*, vol. 33, pp. 9459–9474, 2020.

[6] Jinhao Jiang, Kun Zhou, Zican Dong, Keming Ye, Wayne Xin Zhao, and Ji-Rong Wen, "Structgpt: A general framework for large language model to reason over structured data," *arXiv preprint arXiv:2305.09645*, 2023.

[7] Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D Manning, "Raptor: Recursive abstractive processing for tree-organized retrieval," in *The Twelfth International Conference on Learning Representations*, 2024.

[8] Laura Caspari, Kanishka Ghosh Dastidar, Saber Zerhoudi, Jelena Mitrovic, and Michael Granitzer, "Beyond benchmarks: Evaluating embedding model similarity for retrieval augmented generation systems," *arXiv preprint arXiv:2407.08275*, 2024.

[9] Garima Agrawal, Tharindu Kumarage, Zeyad Alghamdi, and Huan Liu, "Mindful-rag: A study of points of failure in retrieval augmented generation," in *2024 2nd International Conference on Foundation and Large Language Models (FLLM)*. IEEE, 2024, pp. 607–611.

[10] Chunjing Gan, Dan Yang, Binbin Hu, Hanxiao Zhang, Siyuan Li, Ziqi Liu, Yue Shen, Lin Ju, Zhiqiang Zhang, Jinjie Gu, et al., "Similarity is not all you need: Endowing retrieval augmented generation with multi layered thoughts," *arXiv preprint arXiv:2405.19893*, 2024.

[11] William C Mann and Sandra A Thompson, "Rhetorical structure theory: Toward a functional theory of text organization," *Text - Interdisciplinary Journal for the Study of Discourse*, vol. 8, no. 3, pp. 243–281, 1988.

[12] Aaron van den Oord, Yazhe Li, and Oriol Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[13] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning, "HotpotQA: A dataset for diverse, explainable multi-hop question answering," in *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2018.

[14] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng, "MS MARCO: A human generated machine reading comprehension dataset," *CoRR*, vol. abs/1611.09268, 2016.

[15] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal, " musique: Multihop questions via single-hop question composition," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 539–554, 2022.

[16] 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, "Qwen3 embedding: Advancing text embedding and reranking through foundation models," *arXiv preprint arXiv:2506.05176*, 2025.

[17] Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu, "Bge m3-embedding: Multilingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation," 2024.