DON'T RETRIEVE, GENERATE: PROMPTING LLMS FOR SYNTHETIC TRAINING DATA IN DENSE RETRIEVAL

Aarush Sinha
Depatment of Computer Science, University of Copenhagen
aarush.sinha@gmail.com

ABSTRACT

Training effective dense retrieval models typically relies on hard negative (HN) examples mined from large document corpora using methods such as BM25 or cross-encoders, which require full corpus access and expensive index construction. We propose generating synthetic hard negatives directly from a provided query and positive passage, using Large Language Models(LLMs). We fine-tune DistilBERT using synthetic negatives generated by four state-of-the-art LLMs ranging from 4B to 30B parameters (Qwen3, LLaMA3, Phi4) and evaluate performance across 10 BEIR benchmark datasets. Contrary to the prevailing assumption that stronger generative models yield better synthetic data, find that our generative pipeline consistently underperforms traditional corpus-based mining strategies (BM25 and Cross-Encoder). Furthermore, we observe that scaling the generator model does not monotonically improve retrieval performance and find that the 14B parameter model outperforms the 30B model and in some settings it is the worst performing.

1 INTRODUCTION

Dense retrieval models, which encode queries and documents into low-dimensional vectors, are foundational to modern information retrieval systems (Karpukhin et al. (2020); Xiong et al. (2020); Reimers & Gurevych (2019)). These models typically require fine-tuning on task-specific data using contrastive objectives like triplet loss, which pulls relevant pairs closer while pushing away irrelevant (negative) passages Henderson et al. (2017).

A key factor in effective fine-tuning is the quality of negative examples (Xiong et al. (2020)). While random negatives are easy to generate, hard negatives semantically related but irrelevant passages provide stronger learning signals (Zhan et al. (2021)). Traditionally, these are mined from the corpus using methods like BM25 or neural cross-encoders (CEs) for re-ranking (Nogueira & Cho (2020); Qu et al. (2021)).

Large Language Models (Brown et al. (2020); Grattafiori et al. (2024); Kavukcuoglu (2025)) have become exceedingly good at understanding and generating text by following user instructions. We propose an alternative, corpus-free pipeline that utilizes the generative abilities of LLMs. Our approach includes:

1. **Hard Negative Generation:** The LLMs are prompted to produce five semantically challenging but irrelevant passages using 4 LLMs Qwen3-4B & Qwen3-30B(Yang et al. (2025)), LLama3-8B (AI@Meta (2024)) and Phi4-14B (Abdin et al. (2024)).

We evaluate an end-to-end LLM-HN approach against two corpus-based baselines, BM25-HN and CE-HN, by fine-tuning a vanilla DistilBERT(Sanh et al. (2020)) and measuring retrieval performance on 10 BEIR tasks(Thakur et al. (2021)). We analyze the impact of LLM-generated hard negatives and additionally consider a naive combination of BM25-, cross-encoder, and LLM-based hard negatives for training, with all models evaluated on BEIR.

2 Related Work

Large neural models have been increasingly applied across the information retrieval pipeline. For query understanding, techniques like query expansion or generating hypothetical documents aim to improve retrieval effectiveness (Zhu et al. (2023); Ding et al. (2024)). Generative models also serve as powerful re-ranking components, refining initial candidate lists through pairwise or listwise comparisons (Ma et al. (2023); Qin et al. (2024)).

A significant area of research involves using generative models to create synthetic data for training dense retrievers, particularly valuable when labeled data is scarce. Approaches like InPars (Bonifacio et al. (2022)) generate relevant documents for given queries, while others focus on unsupervised data augmentation (Meng et al. (2023)) or specialized data for conversational contexts (Chen et al. (2024)) or training smaller models (Tamber et al. (2025)).

Most closely related to our work are approaches that use generative models to synthesize hard negative samples. SyNeg (Wang et al. (2024)) leverages multi-attribute prompting and reflection and often combines synthetic negatives with corpus-mined ones. Other studies focus on optimizing negative sampling distributions(Ma et al. (2024)), exploiting citation networks or knowledge graphs(Sinha et al. (2025)), or generating negatives for specific settings such as conversational retrieval(Jin et al. (2023)).

3 Methodology

The initial corpus comprises 10,000 randomly sampled abstracts from ms-marco (Bajaj et al. (2018)). We utilize the `Tevatron/msmarco-passage`[1] which already contains queries and positive passages.

BM25 Indexing and Hard Negative Retrieval: Following the construction of a unique passage corpus and its tokenization using NLTK(Bird & Loper (2004)) BM25[2] was initialized. This model indexed the tokenized corpus, computing necessary term statistics such as inverse document frequency (IDF). For each query-passage pair (q, p^+) in the dataset, where q is the query and p^+ is its ground-truth relevant passage, hard negative passages were retrieved.

Hard Negative Mining using Cross-Encoders: For each query-passage pair (q, p^+) in the dataset, where q is the query and p^+ is its ground-truth relevant passage, hard negative passages were retrieved using the `msmarco-MiniLM-L6-v3`. These models were initialized with help of sentence similarity (Reimers & Gurevych (2019)). Cosine similarity was used as the scoring function, and follows the procedure as previous works (Wang et al. (2021)).

Hard-Negatives Generation using LLMs: The goal is to leverage the LLM's understanding and generative capabilities to create passages that appear relevant to a given query but do not actually answer it, serving as highly challenging negative examples for information retrieval tasks. We detail the inference setup in Appendix A. A structured chat prompt was engineered to guide the LLM in generating hard negative passages. The prompts consist of a system instruction and a user query template which are provided in Appendix B.

4 Experiments

4.1 Fine-Tuning

We fine-tune DistilBERT(Sanh et al. (2020)) for a single epoch using a 90/10 train-validation split, with early stopping applied after three consecutive evaluation steps without improvement. Model evaluation is performed every 100 training steps. We conduct separate fine-tuning runs for each set of hard negatives generated by the four LLMs, as well as for those obtained via BM25 and the cross-encoder. In addition, we perform fine-tuning on datasets constructed by naively concatenating

[1] `https://huggingface.co/datasets/Tevatron/msmarco-passage`

[2] `https://pypi.org/project/rank-bm25/`

Fine-Tuning Dataset	FEVER	NFCorpus	SciDocs	SciFact	COVID	NQ	Climate-FEVER	ArguAna	Quora	FiQA	Avg
Baselines											
BM25	**0.610**	0.220	0.116	0.407	0.328	0.278	**0.183**	0.445	0.811	0.190	0.359
Cross-encoder (CE)	0.575	0.225	**0.121**	**0.447**	0.344	**0.298**	0.166	**0.466**	0.815	**0.198**	**0.366**
Aggregated (All LLMs)											
All LLMs	0.415	**0.235**	0.106	0.264	0.265	0.199	0.101	0.381	0.796	0.161	0.292
All LLMs + BM25	0.542	0.222	0.114	0.364	0.345	0.260	0.149	0.422	0.818	0.195	0.343
All LLMs + Cross-Encoder	0.397	0.213	0.112	0.250	0.318	0.284	0.063	0.449	0.816	0.169	0.307
All LLMs + BM25 + Cross-Encoder	0.545	0.220	0.117	0.355	0.296	0.281	0.120	0.479	0.821	0.191	0.342
BM25 + Cross-Encoder + LLM											
BM25 + CE + LLaMA3-8B	0.501	0.202	0.080	0.329	0.279	0.184	0.129	0.295	0.822	0.192	0.301
BM25 + CE + Phi4-14B	0.458	0.196	0.064	0.267	0.255	0.151	0.061	0.280	0.817	0.174	0.272
BM25 + CE + Qwen-4B	0.446	0.190	0.083	0.276	0.254	0.160	0.096	0.215	0.818	0.180	0.272
BM25 + CE + Qwen3-30B	0.326	0.192	0.075	0.302	0.286	0.147	0.067	0.325	0.813	0.185	0.272
BM25 + LLM											
BM25 + Phi4-14B	0.525	0.231	0.109	0.387	0.275	0.184	0.144	0.409	0.800	0.161	0.343
BM25 + LLaMA3-8B	0.518	0.221	0.102	0.413	0.311	0.232	0.172	0.400	0.814	0.173	0.336
BM25 + Qwen-4B	0.577	0.212	0.106	0.367	0.306	0.270	0.119	0.421	0.815	0.182	0.338
BM25 + Qwen-30B	0.542	0.207	0.099	0.321	0.307	0.255	0.085	0.406	0.813	0.165	0.320
Cross-Encoder (CE) + LLM											
CE + Phi4-14B	0.535	0.221	0.109	0.421	**0.357**	0.293	0.150	0.453	**0.823**	0.190	0.355
CE + LLaMA3-8B	0.555	0.222	0.105	0.424	0.301	0.296	0.158	0.403	0.819	0.174	0.346
CE + Qwen-4B	0.470	0.203	0.109	0.330	0.269	0.296	0.089	0.439	0.815	0.177	0.320
CE + Qwen3-30B	0.527	0.211	0.103	0.349	0.297	0.278	0.099	0.422	0.812	0.171	0.327
Standalone LLMs											
Phi4-14B	0.439	0.230	0.109	0.387	0.275	0.184	0.144	0.409	0.800	0.161	0.314
LLaMA3-8B	0.214	0.200	0.056	0.331	0.277	0.102	0.080	0.349	0.793	0.068	0.247
Qwen-4B	0.251	0.223	0.118	0.237	0.328	0.211	0.091	0.414	0.795	0.148	0.282
Qwen-30B	0.317	0.217	0.096	0.377	0.262	0.160	0.124	0.398	0.798	0.145	0.290

Table 1: NDCG@10 results across BEIR datasets on fine-tuning DistilBERT. **Bold** denotes the best score.

the LLM-generated hard negatives with those produced by BM25 and the cross-encoder, resulting in a total of twelve distinct fine-tuning datasets.

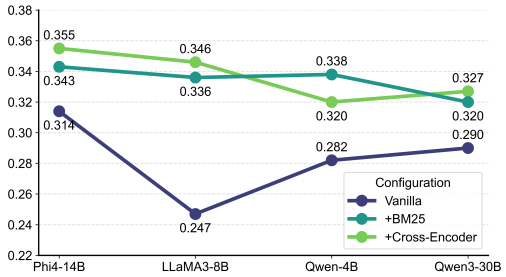
All models were fine-tuned using the Multiple-Negative Ranking Loss (MNRL)(Henderson et al. (2017)), with `batch_size=16`.

4.2 RESULTS

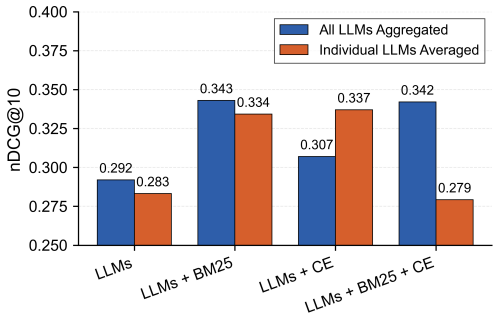
Overall Performance: We evaluate retrieval performance on 10 BEIR Thakur et al. (2021) datasets using nDCG@10. As shown in Table 1, classical retrieval baselines outperform all LLM-only settings. BM25 achieves an average nDCG@10 of **0.359**, while the cross-encoder baseline performs best overall with an average score of **0.366**. In contrast, models fine-tuned solely with LLM-generated hard negatives achieve substantially lower performance, with an average nDCG@10 of **0.292**, indicating that synthetic negatives alone are insufficient to match traditional retrieval supervision.

LLM Performance: We generate hard negatives using four LLMs with sizes ranging from 4B to 30B parameters. As seen in Figure 1a performance does not increase monotonically with model size. The best standalone LLM is Phi4-14B, achieving an average nDCG@10 of **0.314**, while the 30B model reaches **0.290**. Notably, the Qwen3-4B model achieves **0.282**, outperforming the 8B model (**0.247**). These results suggest that model scale alone does not determine the effectiveness of generated hard negatives.

Concatenation of Datasets: We construct twelve datasets by naively concatenating LLM-generated hard negatives with each other and with BM25 and cross-encoder–mined hard negatives.



(a) Average nDCG@10 on the 10 BEIR datasets for DistilBERT fine-tuned using LLM-generated hard negatives, versus fine-tuning with LLM hard negatives combined with BM25- or cross-encoder-mined hard negatives.



(b) Average nDCG@10 across BEIR datasets comparing (i) all LLM-generated hard negatives aggregated, and (ii) performance averaged over individual LLMs, for vanilla LLM negatives, LLM + BM25, and LLM + cross-encoder (CE) mining.

Figure 1: Performance analysis of DistilBERT fine-tuned on synthetic hard negatives.

As shown in Table1, adding both BM25 and cross-encoder hard negatives provides a stronger training signal than LLM negatives alone, yet surprisingly, the traditional BM25 or cross-encoder supervision baselines still yield better retrieval performance overall. Notably, the LLaMA3-8B + CE dataset which incorporates negatives from the LLM that performed worst in isolation yields the second-best average performance within the CE + LLM configurations. Furthermore, fine-tuning DistilBERT on Phi4-14B + CE achieves the highest score on COVID(**0.357**) and Quora (**0.823**), surpassing the baselines, while the Qwen3-30B dataset leads to the lowest performance when concatenated with BM25 negatives. This underscores that larger LLMs do not necessarily generate more effective hard negative training signals.

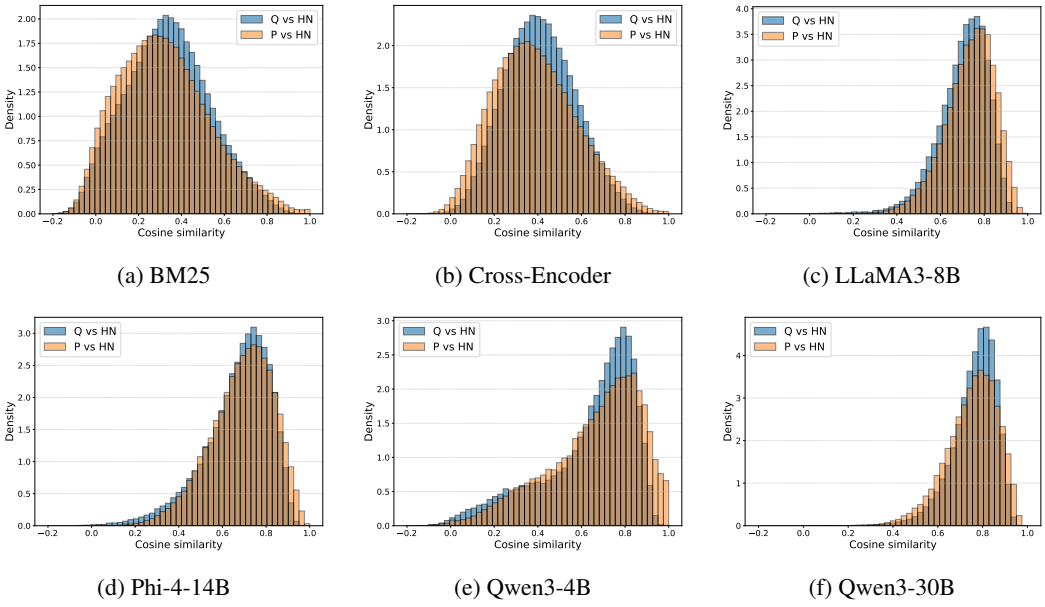
Impact of Aggregation: Figure1b compares the retrieval performance of a dense retriever fine-tuned on a dataset formed by concatenating hard negatives from all LLMs against the average performance of retrievers fine-tuned on individual LLM datasets. We observe that concatenating LLM-generated hard negatives leads to higher average performance than using individual LLM-generated datasets when combined with BM25, BM25+CE and the concatenated dataset as is; we also see that on NFCorpus(**0.235**) the "All LLM" dataset is the best performing . Moreover, incorporating both BM25 and cross-encoder hard negatives substantially improves retrieval performance compared to dense retrievers trained on LLM-generated negatives alone. However, with the exception of LLaMA3-8B, we observe a performance degradation when combining signals: fine-tuning on the full combination of BM25, cross-encoder, and LLM hard negatives results in worse retrieval performance than fine-tuning on the standalone LLM-generated hard negatives.

4.3 Analysis of Generated Hard-Negatives

In Figure 2, we examine the cosine similarity distributions of the generated and mined hard negatives against the corresponding queries (Q) and positive documents (P). A consistent trend emerges across all LLM configurations: their similarity distributions are heavily concentrated near the upper bound (1.0). This implies that the LLMs are generating hard negatives that are overly similar to the ground-truth contexts, potentially bordering on false negatives. Conversely, the distributions for BM25 and cross-encoders span a significantly wider and more even range of similarities. These findings suggest that the superior performance of corpus-mined baselines stems from their ability to expose the retriever to a broader spectrum of negative signals, whereas a lack of variance in LLM-generated negatives limits the contrastive learning process.

5 Conclusions

We proposed a corpus-free pipeline for training dense retrievers by prompting LLMs to generate hard negatives, eliminating the need for full-corpus access during the mining phase. Our evaluation



(a) BM25 (b) Cross-Encoder (c) LLaMA3-8B

(d) Phi-4-14B (e) Qwen3-4B (f) Qwen3-30B

Figure 2: Cosine similarity distribution of generated/mined hard negatives (HN) against the queries (Q) and positives (P).

on the BEIR benchmark reveals that while LLM-generated negatives provide a viable training signal, they currently yield lower retrieval performance compared to traditional BM25 and cross-encoder mining strategies. Furthermore, we find that LLM scale is not a decisive factor in the quality of generated negatives, as smaller models such as Phi4-14B often outperformed larger counterparts like Qwen3-30B. Critically, our experiments with naive dataset concatenation demonstrate that simply combining synthetic and retrieved negatives can degrade performance or improve it. This inconsistency suggests that future research should prioritize developing advanced filtering or integration strategies to effectively leverage the complementary strengths of generative and retrieval-based supervision.

REFERENCES

Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Xin Wang, Rachel Ward, Yue Wu, Dingli Yu, Cyril Zhang, and Yi Zhang. Phi-4 technical report, 2024. URL https://arxiv.org/abs/2412.08905.

AI@Meta. Llama 3 model card. 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, Mir Rosenberg, Xia Song, Alina Stoica, Saurabh Tiwary, and Tong Wang. Ms marco: A human generated machine reading comprehension dataset, 2018. URL https://arxiv.org/abs/1611.09268.

Steven Bird and Edward Loper. NLTK: The natural language toolkit. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, pp. 214–217, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/P04-3031/.

Luiz Bonifacio, Noga Alon, Omer Levy, and Ido Dagan. Inpars: Data augmentation for information retrieval using large language models. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2341–2351, 2022.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners, 2020. URL https://arxiv.org/abs/2005.14165.

Haonan Chen, Zhicheng Dou, Kelong Mao, Jiongnan Liu, and Ziliang Zhao. Generalizing conversational dense retrieval via llm-cognition data augmentation, 2024. URL https://arxiv.org/abs/2402.07092.

Yujuan Ding, Wenqi Fan, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. A survey on rag meets llms: Towards retrieval-augmented large language models, 2024.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, and Abhishek Kadian... The llama 3 herd of models, 2024. URL https://arxiv.org/abs/2407.21783.

Matthew Henderson, Rami Al-Rfou, Brian Strope, Yun hsuan Sung, Laszlo Lukacs, Ruiqi Guo, Sanjiv Kumar, Balint Miklos, and Ray Kurzweil. Efficient natural language response suggestion for smart reply, 2017. URL https://arxiv.org/abs/1705.00652.

Zhuoran Jin, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. InstructoR: Instructing unsupervised conversational dense retrieval with large language models. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 6649–6675, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.443. URL https://aclanthology.org/2023.findings-emnlp.443/.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.550. URL https://aclanthology.org/2020.emnlp-main.550/.

Koray Kavukcuoglu. Gemini 2.5: Our newest gemini model with thinking, 3 2025. URL https://blog.google/technology/google-deepmind/gemini-model-thinking-updates-march-2025/#gemini-2-5-thinking. [Online; accessed 2025-04-15].

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Guangyuan Ma, Yongliang Ma, Xing Wu, Zhenpeng Su, Ming Zhou, and Songlin Hu. Task-level distributionally robust optimization for large language model-based dense retrieval, 2024. URL https://arxiv.org/abs/2408.10613.

Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. Zero-shot listwise document reranking with a large language model, 2023. URL https://arxiv.org/abs/2305.02156.

Yao Meng, Chenyan Xiong, Zhenghao Liu, Zhiyuan Liu, and Jiawei Han. Augtriever: Unsupervised dense retrieval by scalable data augmentation, 2023.

Rodrigo Nogueira and Kyunghyun Cho. Passage re-ranking with bert, 2020. URL https://arxiv.org/abs/1901.04085.

Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. Large language models are effective text rankers with pairwise ranking prompting. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Findings of the Association for Computational Linguistics: NAACL*

2024, pp. 1504–1518, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-naacl.97. URL https://aclanthology.org/2024.findings-naacl.97/.

Yingqi Qu, Yuchen Ding, Jing Liu, Kai Liu, Ruiyang Ren, Wayne Xin Zhao, Daxiang Dong, Hua Wu, and Haifeng Wang. RocketQA: An optimized training approach to dense passage retrieval for open-domain question answering. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tur, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou (eds.), *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5835–5847, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.466. URL https://aclanthology.org/2021.naacl-main.466/.

Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (eds.), *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1410. URL https://aclanthology.org/D19-1410/.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, 2020. URL https://arxiv.org/abs/1910.01108.

Aarush Sinha, Pavan Kumar S, Roshan Balaji, and Nirav Pravinbhai Bhatt. Bica: Effective biomedical dense retrieval with citation-aware hard negatives, 2025. URL https://arxiv.org/abs/2511.08029.

Manveer Singh Tamber, Suleman Kazi, Vivek Sourabh, and Jimmy Lin. DRAMA: Diverse augmentation from large language models to smaller dense retrievers, 2025.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models, 2021. URL https://arxiv.org/abs/2104.08663.

Haonan Wang, Zhiyuan Huang, Yifan Gao, Yifan Deng, Can Ma, and Jianfeng Gao. SyNeg: Synthesizing hard negatives from large language models for dense retrieval, 2024.

Kexin Wang, Nandan Thakur, Nils Reimers, and Iryna Gurevych. Gpl: Generative pseudo labeling for unsupervised domain adaptation of dense retrieval. *arXiv preprint arXiv:2112.07577*, 4 2021. URL https://arxiv.org/abs/2112.07577.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. Approximate nearest neighbor negative contrastive learning for dense text retrieval, 2020. URL https://arxiv.org/abs/2007.00808.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025. URL https://arxiv.org/abs/2505.09388.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. Optimizing dense retrieval model training with hard negatives. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, pp. 1503–1512, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450380379. doi: 10.1145/3404835.3462880. URL https://doi.org/10.1145/3404835.3462880.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zheng Liu, Zhicheng Dou, and Ji-Rong Wen. Large language models for information retrieval: A survey, 2023.

A LLM CONFIGURATION AND INFERENCE

The model was loaded using `vllm` Kwon et al. (2023) for efficient inference, and all models were configured

- **Sampling parameters:**
  - Temperature: 0.6
  - Top-p: 0.95
  - Top-k: 20
  - Minimum p: 0.0
  - Maximum tokens: 1024
- Tensor parallel size: 2 & 6 (for the Qwen3-30B model)
- d_type: `float32`
- GPU memory utilization: 0.80

B PROMPTS

B.0.1 USER PROMPT TEMPLATE

The user prompt template provides specific instructions for generating 5 passages, including length constraints and the required output format:

System Prompt for Hard Negative Generation

```
You are an assistant that generates hard negative passages
for information retrieval tasks.  A hard negative is a
passage that seems relevant to the query but does not
actually answer it or provide the correct information.
You will be given both a query and a positive passage (the
correct answer).  Use this context to generate hard negatives
that are semantically similar but factually incorrect or
irrelevant.
```

User Prompt Template for Hard Negative Generation

Generate 5 hard negative passages for the following query and positive passage pair.
The hard negatives should be similar in style and topic to the positive passage but should NOT correctly answer the query. Each passage should be max of 100 words but no less than 75.
Query: {query}
Positive Passage (correct answer): {positive}
Generate 5 hard negative passages that seem relevant but are actually incorrect or don't properly answer the query.
Provide the passages in the following format:
Passage 1: [your first passage]
Passage 2: [your second passage]
Passage 3: [your third passage]
Passage 4: [your fourth passage]
Passage 5: [your fifth passage]

The {query} placeholder is dynamically replaced with the actual query for which negatives are being generated. The LLM's tokenizer was used to apply this chat template, ensuring the correct format for the model.