Scaling Sparse and Dense Retrieval in Decoder-Only LLMs

Hansi Zeng
University of Massachusetts Amherst
Amherst, MA, United States
hzeng@cs.umass.edu

Julian Killingback
University of Massachusetts Amherst
Amherst, MA, United States
jkillingback@cs.umass.edu

Hamed Zamani
University of Massachusetts Amherst
Amherst, MA, United States
zamani@cs.umass.edu

ABSTRACT

Scaling large language models (LLMs) has shown great potential for improving retrieval model performance; however, previous studies have mainly focused on dense retrieval trained with contrastive loss (CL), neglecting the scaling behavior of other retrieval paradigms and optimization techniques, such as sparse retrieval and knowledge distillation (KD). In this work, we conduct a systematic comparative study on how different retrieval paradigms (sparse vs. dense) and fine-tuning objectives (CL vs. KD vs. their combination) affect retrieval performance across different model scales. Using MSMARCO passages as the training dataset, decoder-only LLMs (Llama-3 series: 1B, 3B, 8B), and a fixed compute budget, we evaluate various training configurations on both in-domain (MSMARCO, TREC DL) and out-of-domain (BEIR) benchmarks. Our key findings reveal that: (1) Scaling behaviors emerge clearly only with CL, where larger models achieve significant performance gains, whereas KD-trained models show minimal improvement, performing similarly across the 1B, 3B, and 8B scales. (2) Sparse retrieval models consistently outperform dense retrieval across both in-domain (MSMARCO, TREC DL) and out-of-domain (BEIR) benchmarks, and they demonstrate greater robustness to imperfect supervised signals. (3) We successfully scale sparse retrieval models with the combination of CL and KD losses at 8B scale, achieving state-of-the-art (SOTA) results in all evaluation sets.

CCS CONCEPTS

• **Information systems** → **Document representation**; **Learning to rank**.

KEYWORDS

Sparse Retrieval; Dense Retrieval; Scaling LLMs

ACM Reference Format:
Hansi Zeng, Julian Killingback, and Hamed Zamani. 2025. Scaling Sparse and Dense Retrieval in Decoder-Only LLMs. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '25), July 13–18, 2025, Padua, Italy.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3726302.3730225

1 INTRODUCTION

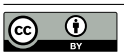
Scaling large language models (LLMs) has led to significant improvements across various NLP tasks [3, 15, 20, 39, 46]. These scaling benefits also extend to retrieval tasks [2, 10, 26, 28, 33, 34]. Recent studies [26, 28, 33] have demonstrated that dense retrieval models parameterized by large decoder-only LLMs, such as LLaMA2-7B [37], can significantly outperform traditional BERT-based retrieval models on both in-domain and out-of-domain benchmarks.

These studies mainly focus on dense retrieval, neglecting sparse retrieval [4, 11, 12, 42]—a single-vector retrieval paradigm that uses high-dimension sparse vectors, where each index represents a vocabulary term. An important feature of sparse retrieval models is that each contextualized token embedding is projected into the vocabulary space, where the model can include information about the original token and do term expansion [7, 12, 21, 35]. To do this well, the model needs to know information about each token when it does the projection or it will not be able to include all the relevant information. This is a problem for decoder-only LLMs as their causal attention masks are designed for generation where the next token is not known. Thus, the embedding used for the projection of the ith token is not aware of the true ith token. To address this shortcoming, we adopt the approach from [2] by replacing the causal mask with a bidirectional mask [8] and doing a masked next-token prediction (MNTP) [2] pre-training step. This modification minimally alters the decoder-only LLM architecture while making it better suited for sparse retrieval. With this issue resolved, we can now systematically compare sparse and dense retrieval in LLMs to understand their scaling behaviors and effectiveness.

Fang et al. [10] demonstrated that scaling dense retrieval models reduces contrastive entropy and improves test-set ranking metrics when trained with contrastive loss (CL). However, the scaling behavior of retrieval models trained with knowledge distillation (KD) [14, 16, 17, 25, 31, 32, 44], another widely used and effective retrieval optimization method, remains underexplored. KD typically employs an external cross-encoder reranker as a teacher model to provide augmented training labels for optimizing student retrieval models [17]. While KD-trained models have been shown to outperform CL-trained models [17, 44], it remains unclear whether this advantage persists as model size increases. As the performance of KD models is inherently tied to the quality of the teacher model, its effectiveness may diminish as models become more capable, making it essential to examine how KD scales with model size.

Furthermore, the scaling behavior of retrieval models in out-of-domain benchmarks was also neglected in previous work [10, 19, 38]. Lin et al. [24] has shown that dense retrieval performance in in-domain and out-of-domain settings is often negatively correlated, whereas sparse retrieval exhibits stronger generalization capabilities. This raises important questions: as retrieval models scale, does this trade-off persist or can scaling mitigate the gap, and whether sparse retrieval continues to generalize better than dense retrieval?

Given these open questions, our work conducts a comprehensive study of how different retrieval paradigms (sparse and dense) and



SIGIR '25, July 13–18, 2025, Padua, Italy

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3730225

Table 1: The overview of experimental setup.

Modeling and Training Objectives:	
LLM Model Sizes	1B, 3B, 8B (Llama3 series)
Retrieval Paradigms	dense retrieval, sparse retrieval
Pretraining Objective	MNTP
Finetuning Objectives	CL, KD, CL+KD
Training and Evaluation Datasets:	
Pretrain and Finetuning Corpus	MSMARCO (8.8M docs, 532K queries)
In-domain Evaluation	MSMARCO-Dev, TREC DL 19 & 20
Out-of-domain Evaluation	BEIR

fine-tuning objectives (CL and KD) interact with model scaling. Specifically, we use the LLaMA3 [29] series (1B, 3B, and 8B) as the backbone for our retrieval models. To ensure a fair comparison, we fix the compute budget in all training configurations. For evaluation, we use MS MARCO Dev [1], TREC DL 2019 [6], and TREC DL 2020 [5] as in-domain benchmarks, and the BEIR benchmark [36] for out-of-domain zero-shot evaluation. We employ parameter-efficient fine-tuning via LoRA [18] for training. Our experiments reveal several key findings:

- Unlike contrastive loss, KD-trained models do not exhibit significant performance gains as model size increases. For instance, KD-trained dense retrieval models overfit and underperform their CL-trained counterparts in BEIR at the 3B and 8B scales.
- Sparse retrieval models consistently outperform dense retrieval models across all evaluation benchmarks with all the fine-tuning objectives. They demonstrate greater robustness as demonstrated by the out-of-domain BEIR benchmark.
- The combination of CL and KD loss achieves the best performance, enabling our 8B sparse retrieval model, named Lion-SP-8B, to become the state-of-the-art model in all the evaluation sets. For instance, Lion-SP-8B outperforms ColBERTv2 by 10.6% in BEIR, and RepLlama by 4.1% in TREC DL 19+20.

To improve reproducibility, we open-source our codebase in https://github.com/HansiZeng/scaling-retriever.

2 SETUP

Our experiments focus on the impact of three variables on in-domain and out-of-domain performance. These variables are: (1) the size of the backbone model. We use the Llama 3 family of models from 1B to 8B (2) the retrieval paradigm including sparse and dense retrieval (3) the fine-tuning objective. The full information on these variables and the training details are presented in Table 1.

2.1 Retrieval Paradigm

In this work, we focus on two widely-adopted retrieval paradigms: dense retrieval [23, 40, 44] and sparse retrieval [11, 12, 42]. These approaches are chosen because they have demonstrated strong retrieval performance [11, 24, 44] and both fall under the category of single-vector retrieval, where each document (or query) is encoded into a single vector.

Given the text $\mathsf{T} = [t_1, \ldots, t_L]$ with L tokens and the large language model (LLM) M_θ we can get the contextualized sequence representation for the text with hidden dimension D as follows:

$$\mathbf{H}_T = M_\theta(\mathsf{T}) \in \mathbb{R}^{D \times L}.$$

For our *dense retrieval* we use mean pooling to obtain a single vector representation:

$$\vec{\mathbf{h}}_T = \text{AvgPool}(\mathbf{H}_T) \in \mathbb{R}^D.$$

For *sparse retrieval*, we project $\mathbf{H}_T$ into the high-dimensional subword token space by doing a matrix multiplication with the LLM's embedding table $\mathbf{E} \in \mathbb{R}^{D \times V}$ where V is the vocabulary size, followed by a max-pooling and rescaling operation similar to [11]:

$$\vec{\mathbf{c}}_T = \log\left(1 + \text{ReLU}\big(\text{MaxPool}(\mathbf{E}^T \cdot \mathbf{H}_T)\big)\right) \in \mathbb{R}^V.$$

To sparsify the high-dimensional vector $\vec{\mathbf{c}}_T$, we apply FLOP regularization [30] during training.

For dense and sparse retrieval the relevance score, $s(\mathsf{q}, \mathsf{d})$, between a query q and a document d is computed in a unified way. Given the corresponding vector representations $\vec{\mathbf{q}}$ and $\vec{\mathbf{d}}$ the relevance score is defined as:

$$s(\mathsf{q}, \mathsf{d}) = \vec{\mathbf{q}} \cdot \vec{\mathbf{d}} \tag{1}$$

2.2 Pretraining

To adapt the decoder-only LLMs for the retrieval tasks, we adopt the pretraining strategy from [2], which consists of two key components: (1) enabling bidirectional attention and (2) applying masked next token prediction (MNTP).

Enabling Bidirectional Attention. We first replace the causal attention mask with a bidirectional attention mask. This allows each token to obtain information from every other token and itself, thus improving the ability to represent the text content and increasing the retrieval performance [2, 33].

Masked Next Token Prediction (MNTP). To adapt the LLMs to bidirectional attention, we train the models with masked next token prediction (MNTP) [2]. Given an input sequence $x = (x_1, x_2, \ldots, x_N)$, we mask 20% of the input tokens and then train the model to predict these masked tokens based on the past and future context. Crucially, when predicting a masked token at position i, we calculate the loss based on the logits obtained from the token representation at the previous position $i-1$, instead of from the masked position itself. This is done because it aligns with the LLM's existing causal training. We use MS MARCO corpus as pre-training corpus.[1]

2.3 Finetuning

We choose (1) Contrastive Loss (CL), (2) Knowledge Distillation (KD), and (3) the combination of CL and KD as our retrieval fine-tuning objectives.

CL is a simple yet effective retrieval training objective. It does not rely on any external models during training, and has been proven effective on both in-domain and out-of-domain evaluation sets [13]. Formally, given a query q, a positive document d^+, and a list of negative documents $\mathcal{D}_q^- := [d_1^-, \ldots, d_N^-]$, using Eq (1), the CL loss for the query q is:

$$\mathcal{L}_{\text{CL}} = -\log \frac{\exp(s(q, d^+))}{\exp(s(q, d^+) + \sum_{d^- \in \mathcal{D}_q^-} \exp(s(q, d^-))}$$

[1] We do at most 10,000 pre-training steps on MS MARCO, which can be completed in 17 hours on 2 A100 GPUs for Llama3-8B.

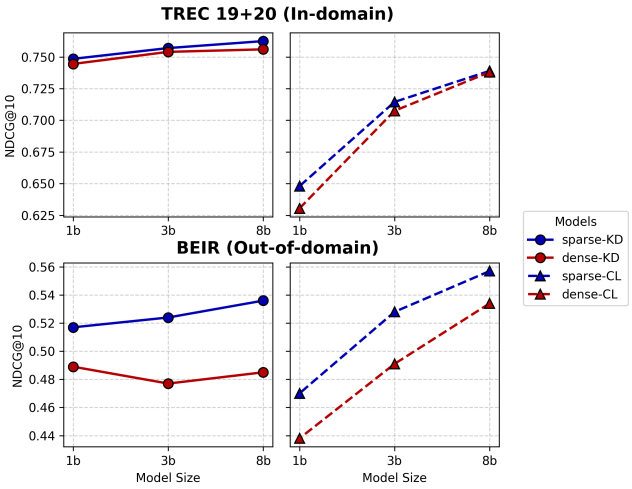


Figure 1: Dense and sparse retrieval results on the combined of TREC DL 19 and 20 (in-domain), and BEIR (out-of-domain) datasets.

KD leverages an external teacher model[2] - typically a cross-encoder reranker - to provide more fine-grained soft labels for retrieval model training. We use MarginMSE [16] as the KD loss in our initial comparison due to its strong performance. Formally, given a query q, a positive document d^+, a negative document d^-, and teacher scores $T(q, d^+)$ and $T(q, d^-)$, the MarginMSE loss is:

$$\mathcal{L}_{KD} = MSE(s(q, d^+) - s(q, d^-),\ T(q, d^+) - T(q, d^-))$$

We also explore combining *CL* and *KD*. As CL works with a list of documents and MarginMSE only works for triplets we switch the KD loss to KL-Divergence [31, 32]. KL-Divergence minimizes the difference in the listwise distribution of the teacher and student. The final loss is the weighted sum of the CL and KD loss terms which is: $\mathcal{L}_{\text{comb}} = 0.5 \cdot (\mathcal{L}_{\text{CL}} + \mathcal{L}_{KD})$.

2.4 Training and Evaluation

We use the MS MARCO passage retrieval dataset [1] for training. For in-domain evaluation, we select MS MARCO Dev [1], TREC Deep Learning (DL) Track 19 and 20 [5, 6], We follow the official evaluation protocol, reporting MRR@10 for MS MARCO Dev and nDCG@10 for TREC DL 19 and 20. For out-of-domain evaluation, we adopt the BEIR benchmark [36]. Following previous work [24, 28, 32], we compute nDCG@10 across 13 datasets in BEIR.

3 ANALYSIS

Scaling Behavior Clearly Emerges with CL, Not with KD. From Figure 1, we observe that when the fine-tuning objective is CL the retrieval model's performance, both for sparse and dense retrieval, increases significantly as the model size grows for both in-domain and out-of-domain evaluation sets.

In contrast, when KD loss is used the performance improvement is far less pronounced with a decrease in performance in some cases. On the in-domain evaluation set, the 1B and 8B KD models perform similarly. Though, compared to CL the KD models perform much better on the in-domain evaluation sets. For example, the 1B KD models outperform the 8B CL models on TREC 19 and 20. However, when evaluated on the out-of-domain BEIR benchmark, KD-models exhibit signs of overfitting. Specifically, dense-KD models show a decline in performance at larger scales. With both sparse and dense KD models performing worse than CL models on out-of-domain evaluation metrics at larger scales. For example, at the 8B scale, both sparse-KD and dense-KD perform worse than their respective sparse-CL and dense-CL models.

[2]We use https://huggingface.co/cross-encoder/ms-marco-MiniLM-L-6-v2.

Table 2: Sparse and dense performance with CL + KD loss. Each decimal is the main metric for the associated evaluation dataset. The first number in parentheses indicates the relative change from only KD while the second is the relative change from only CL.

Model Size	1B	3B	8B
Sparse Retrieval with CL + KD			
MSMARCO Dev	.410 (3.8%, 22.0%)	.417 (5.0%, 7.5%)	.417 (4.2%, 0.5%)
TREC 19+20	.749 (0.1%, 18.9%)	.759 (0.2%, 6.2%)	.762 (-0.1%, 3.2%)
BEIR	.535 (3.5%, 22.1%)	.544 (3.8%, 3.0%)	.552 (3.0%, -0.9%)
Dense Retrieval with CL + KD			
MSMARCO Dev	.404 (4.1%, 20.2%)	.414 (5.6%, 5.9%)	.417 (5.0%, -0.7%)
TREC 19+20	.749 (0.6%, 18.8%)	.760 (0.8%, 7.4%)	.757 (0.2%, 2.7%)
BEIR	.500 (2.2%, 14.2%)	.496 (4.0%, 1.0%)	.501 (3.3%, -6.2%)

Sparse Retrieval consistently Outperforms Dense Retrieval. From Figure 1, we observe that sparse retrieval consistently outperforms dense retrieval with both KD and CL objectives, with a more pronounced increase on the out-of-domain evaluation. For example, at the 8B scale, sparse-KD surpasses dense-KD by 10.5%, while sparse-CL outperforms dense-CL by 4.3% in BEIR.

Moreover, unlike dense-KD, which suffers from overfitting to the teacher model, sparse-KD improves steadily as model size increases. While both paradigms perform similarly on in-domain benchmarks, sparse retrieval demonstrates significantly stronger generalization, achieving much better results in zero-shot settings.

3.1 CL + KD achieves the best Trade-off

From Figure 1, we observe that KD loss significantly boosts the performance of small-scale (1B) retrieval models, while CL provides greater benefits for large-scale (8B) models. We hypothesize that small models benefit from the augmented soft labels provided by the teacher model. However, these soft labels are not perfect, and when the model size reaches 8B, the retrieval model's capacity may surpass that of the cross-encoder teacher model. At this point, CL loss becomes more effective.

Building on these findings, we investigate a combination of CL and KD, named CL + KD, that combines the strengths of both, with results shown in Table 2. We find that CL + KD achieves the best performance balance. For all 1B and 3B models and all evaluation datasets CL + KD increases performance over only KD and only CL. For 8B parameter models there are some decreases in performance but these are offset by gains in other domains in most cases. For instance at the 8B scale, sparse retrieval with the combined loss outperforms the CL-only model by 0.5% on MSMARCO Dev and 3.2% on TREC 19+20, while incurring only a 0.9% loss on BEIR.

Table 3: nDCG@10 of sparse and dense retrieval finetuned with CL, KD or CL +KD objective on the BEIR benchmark. The relative change from the teacher reranked results is shown in the parentheses.

Model Size	1B	3B	8B
Sparse Retrieval in BEIR			
CL	.496 (-8.8%)	.528 (1.3%)	.557 (7.4%)
KD	.517 (0.9%)	.524 (1.9%)	.536 (4.0%)
CL + KD	.535 (4.3%)	.544 (6.1%)	.552 (7.6%)
Dense Retrieval in BEIR			
CL	.435 (-14.7%)	.491 (-5.1%)	.534 (3.1%)
KD	.489 (-4.5%)	.477 (-6.7%)	.485 (-5.3%)
CL + KD	.500 (-1.8%)	.496 (-2.4%)	.501 (-1.4%)

Table 4: Results of the proposed Lion with other SOTA retrieval models. We use paired t-test with Bonferroni correction with p_value < 0.01. The superscripts refer to significant improvements over CL-DRD (∗), SPLADE++ (†), ColBERTv2 (‡), and RepLlama (§).

	In Domain			O.O.D
	MARCO Dev MRR@10	**TREC-19** nDCG@10	**TREC-20** nDCG@10	**BEIR-13** nDCG@10
Model size ≤ 1B				
CL-DRD	.381	.725	.683	.448
SPLADE++	.389	.743	.718	.503
ColBERTv2	.397	.750	.746	.499
Lion-DS-1B	.404*†‡	.757*†‡	.740*†	.500*†
Lion-SP-1B	.410*†‡	.747*†	.751*†	.535*†‡
Model size > 1B				
RepLlama	.412	.743	.721	.551
Lion-DS-8B	**.417***†‡§	.755*†‡§	.759*†‡§	.501*†‡
Lion-SP-8B	**.417***†‡§	**.758***†‡§	**.766***†‡§	**.552***†‡

3.2 Sparse Retrieval might Beat the Teacher

Figure 1 shows that sparse-CL and dense-CL outperform their KD counterparts on BEIR at the 3B and 8B scales. We hypothesize that large retrieval models may have greater capacity than their cross-encoder teacher models. To explore this possibility, we used the teacher model to rerank the original rankings of each retrieval model on BEIR and computed the relative performance difference between the original and reranked results, presented in Table 3.

We observe that sparse-CL surpasses the teacher at both the 3B and 8B scales, while dense-CL outperforms the teacher only at the 8B scale. Another finding is that when KD loss is incorporated (KD and CL + KD), sparse retrieval remains highly robust and consistently outperforms the teacher model. However, for dense retrieval trained with the mixture of losses, performance consistently falls behind the teacher reranker by 1.8%, 2.4%, and 1.4% at the 1B, 3B, and 8B scales, respectively.

4 PERFORMANCE COMPARISON WITH OTHER RETRIEVAL MODELS

For comparison with our 1B models we select state-of-the-art baselines from each retrieval paradigm that are smaller than 1B parameters: CL-DRD [44] (dense retrieval), SPLADE++ [11] (sparse retrieval), and ColBERTv2 [32] (multi-vector retrieval). For models larger than 1B, we use RepLLaMA [28] as the baseline. We refer to our models trained with KD + CL as Lion-[TYPE]-[SIZE] where the type SP indicates a sparse model and DS indicates a dense model, e.g. Lion-SP-8B is the sparse model with LLaMA3-8B as the base model. The in-domain and out-of-domain performance of all models is shown in Table 4.

We observe that Lion-SP-1B and Lion-DS-1B achieve the best overall results among baselines with ≤ 1B parameters, demonstrating that recent large decoder-only LLMs serve as a powerful backbone for document retrieval. For instance, Lion-SP-1B surpasses ColBERTv2 by 3.2% on MS MARCO Dev and by 7.2% on BEIR. Moreover, Lion-SP-8B outperforms the state-of-the-art model RepLLaMA across both in-domain and out-of-domain benchmarks, underscoring the effectiveness of the sparse retrieval paradigm and the combination of CL and KD loss.

5 RELATED WORK

Single-vector retrieval models can be broadly categorized into two types: dense retrieval [16, 17, 23, 26, 28, 31, 33, 40, 43, 44] and sparse retrieval [4, 9, 11, 12, 41, 42]. The key distinction lies in their representations: dense retrieval encodes texts into low-dimensional vectors (i.e. 768 dim in BERT) and employs approximate nearest neighbor (ANN) [22, 40] search for efficient retrieval. In contrast, sparse retrieval represents texts using high-dimensional sparse vectors and relies on an inverted index [4, 47] for fast retrieval.

The most straightforward fine-tuning objective for retrieval models is supervised contrastive loss (CL) [13, 28]. By incorporating techniques such as hard negative sampling [40, 45] and retrieval-oriented continued pre-training [13, 27], the performance of retrieval models can be further improved. Knowledge distillation (KD) [14, 16, 17, 25, 31, 32, 44] is another widely used fine-tuning objective. It has also been shown to perform comparably to or even better than CL for small-scale retrieval models, particularly on in-domain benchmarks [17, 44].

Scaling LLMs has consistently led to improved performance across various NLP tasks [3, 15, 20, 39, 46]. This trend also extends to retrieval models. Fang et al. [10] demonstrated that scaling BERT-based dense retrieval models enhances retrieval performance. Recent works [2, 9, 26, 33] have further explored this trend by training dense retrieval models with larger backbones, such as LLaMA2-7B [37] that achieved state-of-the-art performance.

6 CONCLUSIONS AND FUTURE WORK

We conduct a comprehensive study on the scaling behaviors of different retrieval paradigms and fine-tuning objectives in decoder-only LLMs. One limitation of this work is that we only evaluate knowledge distillation (KD) with a single teacher model. In future work, we aim to expand this analysis by exploring teacher models with varying sizes and different supervision strategies to gain deeper insights into their impact on retrieval performance.

Acknowledgments. This work was supported in part by the Center for Intelligent Information Retrieval, and in part by the Office of Naval Research contract number N000142412612. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

REFERENCES

[1] Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268* (2016).

[2] Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. LLM2Vec: Large Language Models Are Secretly Powerful Text Encoders. In *First Conference on Language Modeling*. https://openreview.net/forum?id=IW1PR7vEBf

[3] 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. 2020. Language Models are Few-Shot Learners. arXiv:2005.14165 [cs.CL] https://arxiv.org/abs/2005.14165

[4] Eunseong Choi, Sunkyung Lee, Minjin Choi, Hyeseon Ko, Young-In Song, and Jongwuk Lee. 2022. SpaDE: Improving Sparse Representations using a Dual Document Encoder for First-stage Retrieval. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management* (Atlanta, GA, USA) *(CIKM '22)*. Association for Computing Machinery, New York, NY, USA, 272–282. https://doi.org/10.1145/3511808.3557456

[5] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, and Daniel Campos. 2021. Overview of the TREC 2020 deep learning track. arXiv:2102.07662 [cs.IR] https://arxiv.org/abs/2102.07662

[6] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. Overview of the TREC 2019 deep learning track. arXiv:2003.07820 [cs.IR] https://arxiv.org/abs/2003.07820

[7] Zhuyun Dai and Jamie Callan. 2020. Context-Aware Term Weighting For First Stage Passage Retrieval. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 1533–1536. https://doi.org/10.1145/3397271.3401204

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *North American Chapter of the Association for Computational Linguistics*. https://api.semanticscholar.org/CorpusID:52967399

[9] Meet Doshi, Vishwajeet Kumar, Rudra Murthy, P Vignesh, and Jaydeep Sen. 2024. Mistral-SPLADE: LLMs for better Learned Sparse Retrieval. *ArXiv* abs/2408.11119 (2024). https://api.semanticscholar.org/CorpusID:271915981

[10] Yan Fang, Jingtao Zhan, Qingyao Ai, Jiaxin Mao, Weihang Su, Jia Chen, and Yiqun Liu. 2024. Scaling Laws For Dense Retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington DC, USA) *(SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 1339–1349. https://doi.org/10.1145/3626772.3657743

[11] Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. 2022. From Distillation to Hard Negative Sampling: Making Sparse Neural IR Models More Effective. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Madrid, Spain) *(SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 2353–2359. https://doi.org/10.1145/3477495.3531857

[12] Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. 2021. SPLADE: Sparse Lexical and Expansion Model for First Stage Ranking. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 2288–2292. https://doi.org/10.1145/3404835.3463098

[13] Luyu Gao and Jamie Callan. 2022. Unsupervised Corpus Aware Language Model Pre-training for Dense Passage Retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (Eds.). Association for Computational Linguistics, Dublin, Ireland, 2843–2853. https://doi.org/10.18653/v1/2022.acl-long.203

[14] Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the Knowledge in a Neural Network. *ArXiv* abs/1503.02531 (2015). https://api.semanticscholar.org/CorpusID:7200347

[15] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Oriol Vinyals, Jack W. Rae, and Laurent Sifre. 2022. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems* (New Orleans, LA, USA) *(NIPS '22)*. Curran Associates Inc., Red Hook, NY, USA, Article 2176, 15 pages.

[16] Sebastian Hofstätter, Sophia Althammer, Michael Schröder, Mete Sertkan, and Allan Hanbury. 2020. Improving Efficient Neural Ranking Models with Cross-Architecture Knowledge Distillation. *ArXiv* abs/2010.02666 (2020). https://api.semanticscholar.org/CorpusID:222141041

[17] Sebastian Hofstätter, Sheng-Chieh Lin, Jheng-Hong Yang, Jimmy Lin, and Allan Hanbury. 2021. Efficiently Teaching an Effective Dense Retriever with Balanced Topic Aware Sampling. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 113–122. https://doi.org/10.1145/3404835.3462891

[18] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*. https://openreview.net/forum?id=nZeVKeeFYf9

[19] Suyuan Huang, Chao Zhang, Yuanyuan Wu, Haoxin Zhang, Yuan Wang, Maolin Wang, Shaosheng Cao, Tong Xu, Xiangyu Zhao, Zengchang Qin, Yan Gao, Yunhan Bai, Jun Fan, Yao Hu, and Enhong Chen. 2024. ScalingNote: Scaling up Retrievers with Large Language Models for Real-World Dense Retrieval. arXiv:2411.15766 [cs.IR] https://arxiv.org/abs/2411.15766

[20] Berivan Isik, Natalia Ponomareva, Hussein Hazimeh, Dimitris Paparas, Sergei Vassilvitskii, and Sanmi Koyejo. 2025. Scaling Laws for Downstream Task Performance in Machine Translation. In *The Thirteenth International Conference on Learning Representations*. https://openreview.net/forum?id=vPOMTkmSiu

[21] Rolf Jagerman, Honglei Zhuang, Zhen Qin, Xuanhui Wang, and Michael Bendersky. 2023. Query Expansion by Prompting Large Language Models. *ArXiv* abs/2305.03653 (2023). https://api.semanticscholar.org/CorpusID:258546701

[22] Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale similarity search with GPUs. arXiv:1702.08734 [cs.CV] https://arxiv.org/abs/1702.08734

[23] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, Online, 6769–6781. https://doi.org/10.18653/v1/2020.emnlp-main.550

[24] Sheng-Chieh Lin, Akari Asai, Minghan Li, Barlas Oguz, Jimmy Lin, Yashar Mehdad, Wen tau Yih, and Xilun Chen. 2023. How to Train Your Dragon: Diverse Augmentation Towards Generalizable Dense Retrieval. In *The 2023 Conference on Empirical Methods in Natural Language Processing*. https://openreview.net/forum?id=d00kbjbYv2

[25] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. 2021. In-Batch Negatives for Knowledge Distillation with Tightly-Coupled Teachers for Dense Retrieval. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, Anna Rogers, Iacer Calixto, Ivan Vulić, Naomi Saphra, Nora Kassner, Oana-Maria Camburu, Trapit Bansal, and Vered Shwartz (Eds.). Association for Computational Linguistics, Online, 163–173. https://doi.org/10.18653/v1/2021.repl4nlp-1.17

[26] Zheng Liu, Chaofan Li, Shitao Xiao, Yingxia Shao, and Defu Lian. 2024. Llama2Vec: Unsupervised Adaptation of Large Language Models for Dense Retrieval. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 3490–3500. https://doi.org/10.18653/v1/2024.acl-long.191

[27] Xinyu Ma, Jiafeng Guo, Ruqing Zhang, Yixing Fan, Yingyan Li, and Xueqi Cheng. 2021. B-PROP: Bootstrapped Pre-training with Representative Words Prediction for Ad-hoc Retrieval. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, Canada) *(SIGIR '21)*. Association for Computing Machinery, New York, NY, USA, 1513–1522. https://doi.org/10.1145/3404835.3462869

[28] Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2024. Fine-Tuning LLaMA for Multi-Stage Text Retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Washington DC, USA) *(SIGIR '24)*. Association for Computing Machinery, New York, NY, USA, 2421–2425. https://doi.org/10.1145/3626772.3657951

[29] Llama Team AI @ Meta. 2024. The Llama 3 Herd of Models. arXiv:2407.21783 [cs.AI] https://arxiv.org/abs/2407.21783

[30] Biswajit Paria, Chih-Kuan Yeh, Ian E.H. Yen, Ning Xu, Pradeep Ravikumar, and Barnabás Póczos. 2020. Minimizing FLOPs to Learn Efficient Sparse Representations. In *International Conference on Learning Representations*. https://openreview.net/forum?id=SygpC6Ntvr

[31] Ruiyang Ren, Yingqi Qu, Jing Liu, Wayne Xin Zhao, QiaoQiao She, Hua Wu, Haifeng Wang, and Ji-Rong Wen. 2021. RocketQAv2: A Joint Training Method for Dense Passage Retrieval and Passage Re-ranking. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (Eds.). Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 2825–2835. https://doi.org/10.18653/v1/2021.emnlp-main.224

[32] Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. ColBERTv2: Effective and Efficient Retrieval via Lightweight Late

Interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Marine Carpuat, Marie-Catherine de Marneffe, and Ivan Vladimir Meza Ruiz (Eds.). Association for Computational Linguistics, Seattle, United States, 3715–3734. https://doi.org/10.18653/v1/2022.naacl-main.272

[33] Jacob Mitchell Springer, Suhas Kotha, Daniel Fried, Graham Neubig, and Aditi Raghunathan. 2025. Repetition Improves Language Model Embeddings. In *The Thirteenth International Conference on Learning Representations*. https://openreview.net/forum?id=Ahlrf2HGJR

[34] Chongyang Tao, Tao Shen, Shen Gao, Junshuo Zhang, Zhen Li, Zhengwei Tao, and Shuai Ma. 2024. LLMs are Also Effective Embedding Models: An In-depth Overview. *ArXiv* abs/2412.12591 (2024). https://api.semanticscholar.org/CorpusID:274789267

[35] Tao Tao, Xuanhui Wang, Qiaozhu Mei, and ChengXiang Zhai. 2006. Language Model Information Retrieval with Document Expansion. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, Robert C. Moore, Jeff Bilmes, Jennifer Chu-Carroll, and Mark Sanderson (Eds.). Association for Computational Linguistics, New York City, USA, 407–414. https://aclanthology.org/N06-1052/

[36] Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Retrieval Models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*. https://openreview.net/forum?id=wCu6T5xFjeJ

[37] Hugo Touvron et al. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. *ArXiv* abs/2307.09288 (2023). https://api.semanticscholar.org/CorpusID:259950998

[38] Yunli Wang, Zixuan Yang, Zhen Zhang, Zhiqiang Wang, Jian Yang, Shiyang Wen, Peng Jiang, and Kun Gai. 2024. Scaling Laws for Online Advertisement Retrieval. arXiv:2411.13322 [cs.IR] https://arxiv.org/abs/2411.13322

[39] Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, Ed H. Chi, Tatsunori Hashimoto, Oriol Vinyals, Percy Liang, Jeff Dean, and William Fedus. 2022. Emergent Abilities of Large Language Models. *Transactions on Machine Learning Research* (2022). https://openreview.net/forum?id=yzkSU5zdwD Survey Certification.

[40] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021. Approximate Nearest Neighbor Negative Contrastive Learning for Dense Text Retrieval. In *International Conference on Learning Representations*. https://openreview.net/forum?id=zeFrfgyZln

[41] Zhichao Xu, Aosong Feng, Yijun Tian, Haibo Ding, and Lin Leee Cheong. 2025. CSPLADE: Learned Sparse Retrieval with Causal Language Models. https://api.semanticscholar.org/CorpusID:277786967

[42] Hamed Zamani, Mostafa Dehghani, W. Bruce Croft, Erik Learned-Miller, and Jaap Kamps. 2018. From Neural Re-Ranking to Neural Ranking: Learning a Sparse Representation for Inverted Indexing. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management* (Torino, Italy) *(CIKM '18)*. Association for Computing Machinery, New York, NY, USA, 497–506. https://doi.org/10.1145/3269206.3271800

[43] Hansi Zeng, Surya Kallumadi, Zaid Alibadi, Rodrigo Nogueira, and Hamed Zamani. 2023. A Personalized Dense Retrieval Framework for Unified Information Access. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, New York, NY, USA, 121–130. https://doi.org/10.1145/3539618.3591626

[44] Hansi Zeng, Hamed Zamani, and Vishwa Vinay. 2022. Curriculum Learning for Dense Retrieval Distillation *(SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 1979–1983. https://doi.org/10.1145/3477495.3531791

[45] Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, M. Zhang, and Shaoping Ma. 2021. Optimizing Dense Retrieval Model Training with Hard Negatives. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (2021). https://api.semanticscholar.org/CorpusID:233289894

[46] Biao Zhang, Zhongtao Liu, Colin Cherry, and Orhan Firat. 2024. When Scaling Meets LLM Finetuning: The Effect of Data, Model and Finetuning Method. In *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=5HCnKDeTws

[47] Justin Zobel and Alistair Moffat. 2006. Inverted files for text search engines. *ACM Comput. Surv.* 38, 2 (July 2006), 6–es. https://doi.org/10.1145/1132956.1132959