Exploring ℓ_0 Sparsification for Inference-free Sparse Retrievers

Xinjie Shen*
South China University of Technology
School of Future Technology
Guangzhou, China
Amazon
Amazon Web Service
Shanghai, China
202164690138@mail.scut.edu.cn

Zhichao Geng
Amazon
Amazon Web Service
Shanghai, China
zhichaog@amazon.com

Yang Yang
Amazon
Amazon Web Service
Shanghai, China
yych@amazon.com

Abstract

With increasing demands for efficiency, information retrieval has developed a branch of sparse retrieval, further advancing towards inference-free retrieval where the documents are encoded during indexing time and there is no model-inference for queries. Existing sparse retrieval models rely on FLOPS regularization for sparsification, while this mechanism was originally designed for Siamese encoders, it is considered to be suboptimal in inference-free scenarios which is asymmetric. Previous attempts to adapt FLOPS for inference-free scenarios have been limited to rule-based methods, leaving the potential of sparsification approaches for inference-free retrieval models largely unexplored. In this paper, we explore ℓ_0 inspired sparsification manner for inference-free retrievers. Through comprehensive out-of-domain evaluation on the BEIR benchmark, our method achieves state-of-the-art performance among inference-free sparse retrieval models and is comparable to leading Siamese sparse retrieval models. Furthermore, we provide insights into the trade-off between retrieval effectiveness and computational efficiency, demonstrating practical value for real-world applications.

CCS Concepts

• **Information systems** → **Information retrieval**; **Document representation**.

Keywords

SPLADE; Inference-free; FLOPS; Sparse Retriever; Passage Retrieval

ACM Reference Format:

Xinjie Shen, Zhichao Geng, and Yang Yang. 2025. Exploring ℓ_0 Sparsification for Inference-free Sparse Retrievers. 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, 5 pages. https://doi.org/10.1145/3726302.3730192

1 Introduction

Information retrieval systems have evolved significantly over the past decades, with sparse retrieval remaining a cornerstone branch due to its efficiency, scalability, and reasonable search relevance[15]. While dense retrievers such as ColBERTv2[25] and Contriever[14] learn continuous semantic representations of queries and documents through neural networks, sparse retrievers focus on capturing lexical matches through token-based representations[4, 17, 27].

To further enhance retrieval efficiency, inference-free retrieval methods[5, 12, 16, 21] have emerged as a promising solution to the computational challenges in modern search systems[2]. These approaches pre-compute document representations while degenerating query-time inference into lightweight methods such as term matching[23, 24] or tokenization[7, 8, 10, 16]. This design achieves superior efficiency while maintaining competitive performance.

The efficiency of sparse retrieval models is largely determined by their sparsification capability. A key design element in Siamese sparse retriever models is the FLOPS (Floating Point Operations) regularization[22], which penalizes non-zero elements in sparse representations using their squared average. However, this approach is less suited for inference-free models due to their asymmetric architecture, which differs from the original Siamese design in models like SPLADE. Moreover, FLOPS continually penalizes the scale of token weights but pays less attention to tokens' ℓ_0. The encoded document length could lower to zero with larger weight on FLOPS,

In this paper, we propose a combination of two novel approaches designed for sparse inference-free models: ℓ_0 mask FLOPS and ℓ_0 approximation activation. These techniques introduce selective sparsification by applying regularization only to document side representations exceeding desired sparsity thresholds, while allowing already-sparse representations to optimize unrestrictedly for the ranking objective. Through extensive out-of-domain experiments on BEIR, we demonstrate that we can achieve superior performance to representative dense retriever models and inference-free models, and become comparable with state-of-the-art sparse retriever models, while maintaining good computational efficiency.

The key contributions of our work include:

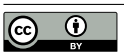
- Two novel sparsification techniques, ℓ_0 mask loss and ℓ_0 approximation activation, designed for inference-free retrievers.
- Strong empirical results on BEIR benchmark, achieving comparable performance to state-of-the-art sparse retrievers.
- Comprehensive analysis of efficiency-performance trade-offs, providing practical insights for real-world scenarios.

2 Related Work

2.1 Sparse Retrievers

Compared to dense retrievers, like ColBERTv2[25], Contriever[14] and TAS-B[13], which learn continuous semantic representations

*Work done during internship at Amazon. Corresponding author: Yang Yang.



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

ACM ISBN 979-8-4007-1592-1/2025/07
https://doi.org/10.1145/3726302.3730192

of both queries and documents utilizing neural networks, sparse representations focus on capture lexical matches through discrete token-based representations[1, 4, 17, 27]. Specifically, sparse representations typically represent documents and queries as high-dimensional sparse vectors where each dimension corresponds to a specific term in the vocabulary, and the value indicates the importance of that term. The SPLADE series models[7, 8, 10, 16] have achieved great success as representative sparse retriever models, utilizing BERT's masked language model head and pooling to generate vocabulary-sized sparse representations. However, query encoding remains an efficiency bottleneck in online practice scenarios, requiring further development and variants for adaptation.

2.2 Inference-free Retrievers

Inference-free retriever models, such as BM25[23, 24], DEEPCT[5], Doc2Query[21], DeepImpact[19], EPIC[18], SPLADE-v3-Doc[16] and SPLADE-doc-distill[7, 12], can directly retrieve relevant documents without requiring computationally expensive neural inference on the query side. These models typically rely on precomputed document representations offline, and simple operations for obtaining query representations online, making them particularly attractive for real-world applications[3, 6] where latency and computational resources are critical concerns due to huge query volumes. Unlike inference-required models that need to encode queries through neural networks during retrieval, inference-free retrievers use simple operations like term matching[23, 24] or tokenization[7, 8, 10, 16] to avoid huge computational costs, while offering lower but reasonable throughput and latency.

3 Preliminary

Our work is built upon the representative sparse inference-free retriever, *SPLADE-doc-distill*. This model predicts token importance across the vocabulary space using BERT's masked language model head. The model creates sparse 30,522-dimensional vectors using max pooling and ReLU activation. It operates asymmetrically, processing only documents. Moreover, we adopt the IDF-aware technique proposed by [12]. For an input document with token weight $(w_{i,j})$ at position (i), the sparse representation of token j is:

$$w_j = \text{IDF}_j \cdot \max_i \log(1 + \sigma(w_{i,j})), \tag{1}$$

where IDF_j is the IDF value of token j and $\sigma(\cdot)$ is the ReLU function.

For query representation, it employs an inference-free approach to generate a Bag-of-Words-like sparse vector, where the weight w_j of token j in the query representation is set to 1 only if token j is present in the query; otherwise, w_j is set to 0. IDF values are then applied for adjustment as well. The ranking score for a given query and document can be computed as the inner product of their sparse representations. The optimization objective of SPLADE combines the ranking loss $\mathcal{L}_{\text{rank}}$ and sparse regularization loss $\mathcal{L}^d_{\texttt{FLOPS}}$ for optimizing search relevance and representation sparsification. For *IDF-SPLADE-doc-distill*, the training loss $\mathcal{L}$ is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{\texttt{rank}} + \lambda_d \mathcal{L}^d_{\texttt{FLOPS}}, \tag{2}$$

$$\mathcal{L}^d_{\texttt{FLOPS}} = \sum_{j \in V} \bar{a}_j^2 = \sum_{j \in V} \left(\frac{1}{N} \sum_{i=1}^{N} \frac{w_j^{(d_i)}}{\text{IDF}_j} \right)^2, \tag{3}$$

where λ_d is the weight for the FLOPS regularizer, d_i is the i-th document in batch and $\bar{a}_j$ is the average weight of token j in the batch. The FLOPS regularizer penalizes dimensions with high average weights in the representation to achieve sparsity. We denoted this base model as *IDF-SPLADE-doc-distill* in the following.

4 Method

4.1 ℓ_0 Mask Loss

Given the loss format in Equation 2 for inference-free sparse retriever, it can be observed that with a relatively large weight of λ_d, the sparsification objective may easily dominate the $\mathcal{L}_{\texttt{rank}}$ objective, pushing the average encoded sparse representations' ℓ_0 to a very limited amount that harms the learning of $\mathcal{L}_{\texttt{rank}}$. In this section, we propose a threshold method named ℓ_0 Mask Loss, which focuses on utilizing the advantages of FLOPS loss in achieving sparsity while avoiding further reduction of token weights and the ℓ_0 of sparse representations (e.g., collapsing to 0), which can harm the learning of the ranking objective, once reasonable sparsity is achieved.

Specifically, we calculate the ℓ_0 of each $w^{(d_i)}$ in the batch and build a binary mask based on a given threshold t. If the amount of activated tokens in $w^{(d_i)}$ is already lower than the given threshold, then $w^{(d_i)}$ will not participate in the calculation of FLOPS loss, allowing free learning without the penalty and constraints on token weight. The $\mathcal{L}^d_{\texttt{FLOPS}}$ can be rewritten as follows:

$$\mathcal{L}^d_{\texttt{FLOPS}}{}' = \sum_{j \in V} \left(\frac{1}{N} \sum_{i=1}^{N} M^{(d_i)} \frac{w_j^{(d_i)}}{\text{IDF}_j} \right)^2, \tag{4}$$

$$M^{(d_i)} = \mathbb{1}[\|w^{(d_i)}\|_0 > t], \tag{5}$$

where $M^{(d_i)}$ is a binary mask indicator vector that equals $\mathbb{1} \in \mathbb{R}^{|V|}$ if the ℓ_0 norm of weight vector $w^{(d_i)}$ exceeds threshold t, and all zeros otherwise. This mask is designed to effectively exclude already-sparse document side presentations from the FLOPS loss calculation, which helps to provide specific focus for the model under asymmetric structure.

4.2 ℓ_0 Approximation Activation

From Equation 3, it can also be observed that FLOPS largely relies on punishing representations via their weights' scale that related to the retrieval time[7], rather than focusing on the ℓ_0 of the representation. This leads to more focus on larger weights and potentially overlooks small weights, since the resulting gradients have diverse differences. To help the regularizer focus more on small weights rather than being overwhelmingly led by large gradients of large weights, we suggest using multiple subsequent log transformations in activation. Compared to the previous design, such modification shifts more attention to the ℓ_0 of the representation rather than the token scale, therefore we named it as ℓ_0 approximation activation here. With applying such modification, larger values grow much slower than a linear function and serve as a soft cap, creating a natural sparsity-inducing effect for the model similar to ℓ_0 regularization.

Code is available at this Github repository: https://github.com/zhichao-aws/opensearch-sparse-model-tuning-sample/tree/l0_enhance. All details are included.

Table 1: Model performances (NCDG@10) on 13 datasets of BEIR. Best performance of each retriever type are bold.

	Inference-free Sparse Retriever						Sparse Retriever		Dense Retriever		
Dataset	ℓ_0 Mask	ℓ_0 Mask-ℓ_0 Activation	IDF-SPLADE-doc-distill	BM25	SPLADE-doc-distill	SPLADE-v3-Doc	SPLADE ++SelfDistil	SPLADE -v3-Distil	ColBERTv2	Contriever	TAS-B
TREC-COVID	70.4	71.3	67.0	68.8	68.4	68.1	71.0	70.0	**73.8**	59.6	48.1
NFCorpus	34.1	34.1	32.6	32.7	34.0	33.8	33.4	**34.8**	33.8	32.8	31.9
NQ	54.0	54.1	52.6	32.6	48.8	52.1	52.1	54.9	**56.2**	49.8	46.3
HotpotQA	67.6	68.1	67.9	60.2	62.6	66.9	**68.4**	67.8	66.7	63.8	58.4
FiQA-2018	35.4	34.6	34.2	25.4	31.2	33.6	33.6	33.9	**35.6**	32.9	30.0
ArguAna	48.9	**49.7**	48.6	47.2	37.7	46.7	47.9	48.4	46.3	44.6	42.9
Touche-2020	29.2	28.3	27.6	34.7	25.6	27.0	**36.4**	30.1	26.3	23.0	16.2
DBPedia-entity	42.2	42.2	41.3	28.7	35.9	36.1	43.5	42.6	**44.6**	34.5	38.4
SCIDOCS	16.1	16.3	16.4	16.5	14.7	15.2	15.8	14.8	15.4	**41.3**	14.9
FEVER	81.0	81.1	**81.6**	64.9	67.4	68.9	78.6	79.6	78.5	16.5	70.0
Climate-FEVER	21.9	21.1	21.5	18.6	15.1	15.9	23.5	22.8	17.6	**75.8**	22.8
SciFact	**71.5**	70.6	70.5	69.0	70.8	68.8	69.3	68.5	69.3	23.7	64.3
Quora	83.1	82.2	82.1	78.9	73.0	77.5	83.8	81.7	85.2	**86.5**	83.5
Aver.Rank	3.54	**3.38**	5.15	7.85	8.62	7.46	**3.92**	4.69	**4.69**	7.38	8.77
Average	**50.43**	50.28	49.52	44.48	45.02	46.97	**50.56**	49.99	**49.95**	44.98	43.67

Since the original design of the SPLADE model has one log, we suggest rewriting the activation function $\sigma(\cdot)$ in Equation 1 as follows:

$$\sigma(x) = \log(1 + \max(0, x)). \tag{6}$$

We replace the original RELU activation function with our ℓ_0 approximation activation during both training and evaluation.

5 Experiment

In this section, we aim to answer two research questions: **Q1**: How do our proposed methods improve performance? **Q2**: How do our methods affect the trade-off between efficiency and performance?

5.1 Experiment Setup

5.1.1 Datasets. The MS MARCO dataset [20] is employed to finetune[1] all models, and we initialize the model from Co-Condenser [11] checkpoint. Teacher scores preparation process follows [12]. The MS MARCO dataset comprises 8,841,823 passages and 502,939 queries in the training set. Following the work of [9, 16], we evaluate our model's **zero-shot** out-of-domain performance on a readily available subset of 13 datasets from the BEIR benchmark [26].

5.1.2 Baselines. In this section, we include three types of baselines: 1) **Inference-free sparse retrievers**: BM25[23, 24], IDF-SPLADE-doc-distill[12], SPLADE-doc-distill[7, 12], SPLADE-v3-Doc[16], 2) **Sparse retrievers**: SPLADE++-SelfDistil[8], SPLADE-v3-Distil[16], 3) **Dense retrievers**: ColBERTv2[25], Contriever[14], TAS-B[13]. During training and evaluation, IDF values are derived from the MS MARCO dataset.

5.1.3 Metrics and Evaluation. We include three metrics to measure models' performance and efficiency: 1) NDCG@10, 2) FLOPS[22], 3) the average number of non-zero tokens in encoded document sparse representation, denoted as Doc_Len. For the NDCG@10 metric, we calculate it using the BEIR python toolkit.

[1]We use the word finetune since models like SPLADE are initialized from BERT pretrained weights

Table 2: Ablation study in comparable FLOPS.

Model	NCDG@10	FLOPS	Doc_Len
Baseline	49.52	2.39	327.23
+ℓ_0 mask	50.43	2.31	322.22
+ℓ_0 Activation	49.97	2.30	295.79
+ℓ_0 mask-ℓ_0 Activation	50.28	2.13	275.02

5.1.4 Indexing. We use OpenSearch[2] as our lexical search engine to construct the inverted index and perform the retrieval process. The writing and searching processes for custom learned sparse models are integrated through the OpenSearch neural sparse feature. The maximum input length is set to 512 tokens.

5.2 (Q1): Relevance Evaluation

In this section, we present the zero-shot out-of-domain (OOD) evaluation results on the BEIR benchmark, as shown in Table 1. Our proposed methods demonstrate significant improvement, outperforming the best inference-free sparse retriever. Moreover, our approach outperforms all dense retrievers and maintains performance comparable to state-of-the-art Siamese sparse retrievers, all while retaining inference-free efficiency.

To assess the overall effectiveness across different datasets, we also analyzed the relative ranking of each method. Our approach achieved the best average rank among all three types of retrievers (dense, inference-free sparse, and Siamese sparse), demonstrating its robust performance across diverse domains and retrieval tasks.

From the ablation study provided in Table 2, we can observe that the ℓ_0 mask remains a good representative with reasonable FLOPS and Doc_Len while maintaining good performance. Application of ℓ_0 Activation explicitly improves the efficiency with limited relevance degradation. However, directly applying ℓ_0 Activation increases efficiency but harms performance, suggesting that ℓ_0 Activation relies on ℓ_0 mask. In the following section, we dive deeper into the trade-off between efficiency and performance.

[2]https://opensearch.org/

5.3 (Q2): Efficiency Trade-off Analysis

5.3.1 Varying FLOPS Penalty. In Figures 1 and 2, we include representative models as baselines and vary hyperparameter λ_d for comparison, with threshold t at 200. The results in Figure 1 show that our proposed methods outperform the baselines in both performance and efficiency. For both ℓ_0 activation and ℓ_0 mask individually, the models achieve greater efficiency at the same performance level. Moreover, the results demonstrate that when applying ℓ_0 activation with ℓ_0 mask together, performance degrades more slowly as efficiency increases.

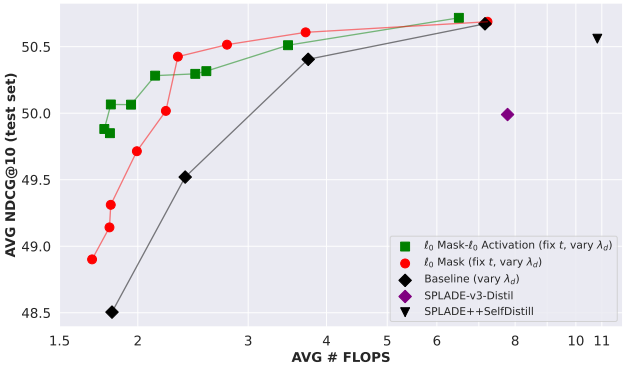


Figure 1: Search relevance vs efficiency, varying λ_d.

In Figure 2, we demonstrate how λ_d affects the resulting Doc_Len. We observed that the encoded Doc_Len of the baseline model collapses to near zero during training when λ_d increases beyond a certain level (0.12 in this case, denoted as "x" in the figure, with other settings fixed), as mentioned in our motivation in Section 4.1. In contrast, our proposed methods remain stable as λ_d increases, making model tuning and selection more flexible. This stability is achieved because the ℓ_0 mask allows encoded documents with lengths below the threshold to avoid further penalty. From the side of Doc_Len, applying ℓ_0 activation helps model achieved encoded document sparsity.

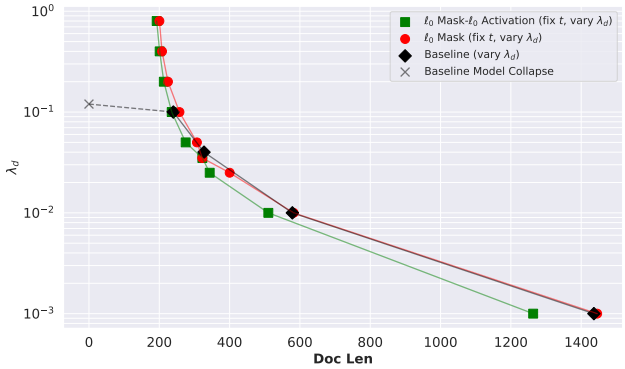


Figure 2: λ_d vs encoded document sparsity. "x" denotes that baseline model collapses during training at $\lambda_d = 0.12$.

5.3.2 Varying Masked Threshold. Furthermore, we discuss the effects of varying the threshold t in our methods, with λ_d fixed at 0.04. In Figure 3, we observe that by varying t in [50, 100, 150, 200, 250, 300, 400 500, 1000], our methods' performance increases when the threshold allows longer documents to avoid penalization. When the threshold approaches zero, the ℓ_0 Mask degenerates to near baseline performance, where FLOPS directly affects all documents. Compared to Figure 1, we can also find that at comparable FLOPS, the results obtained by varying t are higher than those obtained by varying λ_d, indicating more exploration space and flexible model tuning.

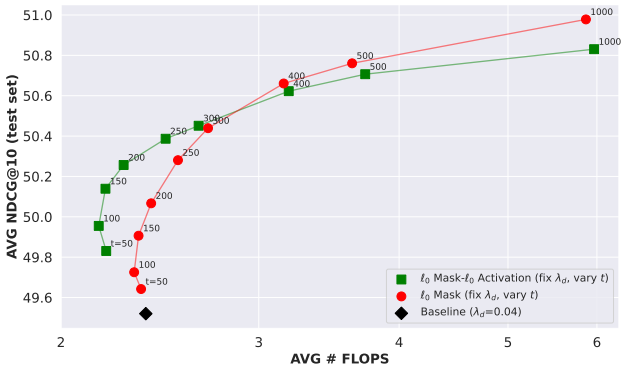


Figure 3: Search relevance vs sparsity, varying t.

5.3.3 Varying ℓ_0 Activation. We investigate the impact of applying ℓ_0 Activation multiple times (from 1 to 3) while keeping other settings fixed. Additionally, We explore two decoupling variants: de-ℓ_0 activation that only uses ℓ_0 activation for FLOPS calculation but remains unchanged for ranking objective, and inv-de-ℓ_0 activation that only uses ℓ_0 activation for ranking objective but remains unchanged for FLOPS calculation. Table 3 shows the metrics for different numbers of ℓ_0 Activation applications and the variant. We find that applying ℓ_0 Activation decreases both FLOPS and Doc_Len, indicating improved efficiency. However, applying it more than once begins to harm performance, which aligns with our motivation and analysis in Section 4.2. Meanwhile, the decoupled versions show similar or lower performance but larger FLOPS and Doc_Len, suggesting we should apply ℓ_0 activation on both ranking objective and FLOPS.

Table 3: Comparison of applying variants of ℓ_0 Activation.

Model (λ_d = 0.035)	NCDG@10	FLOPS	Doc_Len
Baseline	49.52	2.39	327.23
ℓ_0 Activation	49.97	2.30	295.79
de-ℓ_0 Activation	49.77	2.52	345.63
inv-de-ℓ_0 Activation	49.54	2.36	326.58
2-ℓ_0 Activation	49.51	2.12	258.43
3-ℓ_0 Activation	48.83	2.05	245.64
ℓ_0 mask + ℓ_0 Activation	50.28	2.13	275.02
ℓ_0 mask + de-ℓ_0 Activation	50.27	2.61	370.67
ℓ_0 mask + inv-de-ℓ_0 Activation	49.34	2.18	283.78

6 Conclusion

In this paper, we propose simple yet effective ℓ_0-inspired methods that achieve comparable performance to state-of-the-art sparse retrieval models in the inference-free setting, using same and fair training data and evaluation protocols. Moreover, we analyze the trade-off between retrieval effectiveness and computational efficiency, providing practical insights for implementation.

References

[1] Yang Bai, Xiaoguang Li, Gang Wang, Chaoliang Zhang, Lifeng Shang, Jun Xu, Zhaowei Wang, Fangshan Wang, and Qun Liu. 2020. SparTerm: Learning Term-based Sparse Representation for Fast Text Retrieval. https://doi.org/10.48550/ARXIV.2010.00768

[2] Shay Banon, Simon Willnauer, Jason Tedor, Martijn van Groningen, Ryan Ernst, Luca Cavanna, Nik Everett, David Pilato, Adrien Grand, Robert Muir, Boaz Leskes, Clinton Gormley, James Rodewig, Alexander Reelsen, Jay Modi, Lee Hinman, Tanguy Leroux, Colin Goodheart-Smithe, Lisa Cawley, Christoph Büscher, Armin Braun, Igor Motov, Nhat Nguyen, Jim Ferenczi, Yannick Welsch, Dimitris Athanasiou, dependabot[bot], David Roberts, David Turner, and Tal Levy. 2025. *opensearch-project/OpenSearch*. https://github.com/opensearch-project/OpenSearch

[3] Michael Busch, Krishna Gade, Brian Larson, Patrick Lok, Samuel Luckenbill, and Jimmy Lin. 2012. Earlybird: Real-Time Search at Twitter. In *2012 IEEE 28th International Conference on Data Engineering*. 1360–1369. https://doi.org/10.1109/ICDE.2012.149

[4] 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

[5] 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*. 1533–1536.

[6] M. A. H. Dempster, Juho Kanniainen, John Keane, and Erik Vynckier. 2018. *High-Performance Computing in Finance: Problems, Methods, and Solutions* (1st ed.). Chapman & Hall/CRC.

[7] Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. 2021. SPLADE v2: Sparse Lexical and Expansion Model for Information Retrieval. https://doi.org/10.48550/ARXIV.2109.10086

[8] 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

[9] 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

[10] Thibault Formal, Benjamin Piwowarski, and Stéphane Clinchant. 2021. *SPLADE: Sparse Lexical and Expansion Model for First Stage Ranking*. Association for Computing Machinery, New York, NY, USA, 2288–2292. https://doi.org/10.1145/3404835.3463098

[11] 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)*. 2843–2853.

[12] Zhichao Geng, Dongyu Ru, and Yang Yang. 2024. Towards Competitive Search Relevance For Inference-Free Learned Sparse Retrievers. arXiv:2411.04403 [cs.IR] https://arxiv.org/abs/2411.04403

[13] 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

[14] Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Unsupervised Dense Information Retrieval with Contrastive Learning. https://doi.org/10.48550/ARXIV.2112.09118

[15] Carlos Lassance and Stéphane Clinchant. 2022. An Efficiency Study for SPLADE Models. 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, 2220–2226. https://doi.org/10.1145/3477495.3531833

[16] Carlos Lassance, Hervé Déjean, Thibault Formal, and Stéphane Clinchant. 2024. SPLADE-v3: New baselines for SPLADE. https://doi.org/10.48550/ARXIV.2403.06789

[17] Sean MacAvaney, Franco Maria Nardini, Raffaele Perego, Nicola Tonellotto, Nazli Goharian, and Ophir Frieder. 2020. Expansion via Prediction of Importance with Contextualization. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '20)*. ACM, 1573–1576. https://doi.org/10.1145/3397271.3401262

[18] Sean MacAvaney, Franco Maria Nardini, Raffaele Perego, Nicola Tonellotto, Nazli Goharian, and Ophir Frieder. 2020. Expansion via prediction of importance with contextualization. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 1573–1576.

[19] Antonio Mallia, Omar Khattab, Torsten Suel, and Nicola Tonellotto. 2021. Learning Passage Impacts for Inverted Indexes. 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, 1723–1727. https://doi.org/10.1145/3404835.3463030

[20] Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. (November 2016). https://www.microsoft.com/en-us/research/publication/ms-marco-human-generated-machine-reading-comprehension-dataset/

[21] Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019. Document expansion by query prediction. *arXiv preprint arXiv:1904.08375* (2019).

[22] 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

[23] Stephen Robertson and Hugo Zaragoza. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.* 3, 4 (April 2009), 333–389. https://doi.org/10.1561/1500000019

[24] S. E. Robertson and K. Sparck Jones. 1976. Relevance weighting of search terms. *Journal of the American Society for Information Science* 27, 3 (1976), 129–146. https://doi.org/10.1002/asi.4630270302 arXiv:https://asistdl.onlinelibrary.wiley.com/doi/pdf/10.1002/asi.4630270302

[25] 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

[26] 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

[27] Tiancheng Zhao, Xiaopeng Lu, and Kyusong Lee. 2021. SPARTA: Efficient Open-Domain Question Answering via Sparse Transformer Matching Retrieval. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 565–575.

Received 20 February 2007; revised 12 March 2009; accepted 5 June 2009