Neural Lexical Search with Learned Sparse Retrieval

Andrew Yates
University of Amsterdam
Amsterdam, The Netherlands
a.c.yates@uva.nl

Carlos Lassance
Cohere
Grenoble, France
carlos@cohere.com

Sean MacAvaney
University of Glasgow
Glasgow, United Kingdom
sean.macavaney@glasgow.ac.uk

Thong Nguyen
University of Amsterdam
Amsterdam, The Netherlands
t.nguyen2@uva.nl

Yibin Lei
University of Amsterdam
Amsterdam, The Netherlands
y.lei@uva.nl

Abstract

Learned Sparse Retrieval (LSR) techniques use neural machinery to represent queries and documents as learned bags of words. In contrast with other neural retrieval techniques, such as generative retrieval and dense retrieval, LSR has been shown to be a remarkably robust, transferable, and efficient family of methods for retrieving high-quality search results. This half-day tutorial aims to provide an extensive overview of LSR, ranging from its fundamentals to the latest emerging techniques. By the end of the tutorial, attendees will be familiar with the important design decisions of an LSR system, know how to apply them to text and other modalities, and understand the latest techniques for retrieving with them efficiently. Website: https://lsr-tutorial.github.io

CCS Concepts

• **Information systems** → **Retrieval models and ranking**; *Document representation*; *Query representation.*

Keywords

Learned Sparse Retrieval, First-stage Retrieval, Neural Information Retrieval, Content-based Retrieval

ACM Reference Format:
Andrew Yates, Carlos Lassance, Sean MacAvaney, Thong Nguyen, and Yibin Lei. 2024. Neural Lexical Search with Learned Sparse Retrieval. In *Proceedings of the 2024 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region (SIGIR-AP '24), December 9–12, 2024, Tokyo, Japan.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3673791.3698441

1 Extended Abstract

1.1 Motivation and Scope

Neural information retrieval approaches, which use deep learning to improve relevance ranking based on the similarity of query and document content, have greatly improved search quality [31]. These approaches can be divided into two categories: re-ranking methods that use a transformer to compare query and document text at query time and first-stage retrieval methods that produce document representations offline and store them in an index. Re-ranking methods use a pre-trained language model like BERT, T5, or ChatGPT to predict relevance scores at inference time, which is a slow but effective strategy [31, 54]. First-stage retrieval methods produce query and document representations independently, which means that document representations can be pre-computed and only a query representation needs to be computed at query time. While these methods are not as effective as re-ranking methods, they are substantially faster and often used to identify promising candidate documents that can be reranked in a later stage.

First-stage retrieval methods can be divided into generative ranking methods, dense retrieval methods, and learned sparse retrieval methods. *Generative retrieval* is an emerging approach in which document representations are stored in the transformer model itself, removing the need for a separate index but creating new efficiency and scalability challenges [29, 47, 56]. *Dense retrieval* methods, on the other hand, build dense, fixed-dimensional representations for each document, which are then indexed and retrieved over [27]. However, dense retrieval comes with several limitations. Retrieval over the dense vectors does not scale without using approximation techniques like HNSW [37], dense representations can be costly in terms of storage [59], and the latent vector representations used are inherently challenging to interpret [25].

Learned Sparse Retrieval (LSR) techniques provide a solution to these problems by representing documents as sparse vectors, typically with dimensions that represent terms in a vocabulary, akin to traditional Bag-of-Words (BoW) models. However, unlike BoW models, LSR models learn the tokens [35] and weights [16] through training, allowing them to match semantically, as dense and generative retrieval models do. Moreover, their sparsity allows them to leverage efficient posting-list-based retrieval algorithms for fast, exact top-k retrieval [28].

The growing body of work in LSR and its integration into industry search products [24] suggests that LSR has emerged as a prominent and compelling family of retrieval techniques. However, there has not yet been a tutorial focused on this important research direction (unlike other techniques such as Generative Retrieval [55]). This tutorial brings together researchers who have made contributions to LSR—including those in modeling, training, and modality—to cover LSR from its fundamentals to emerging topics. Although the tutorial is directed at those with intermediate experience in IR, we expect the range and depth of topics covered to provide value to beginners and experts too.

SIGIR-AP '24, December 9–12, 2024, Tokyo, Japan

ACM ISBN 979-8-4007-0724-7/24/12
https://doi.org/10.1145/3673791.3698441

1.2 Objectives

After the tutorial, attendees will understand the concepts behind learned sparse retrieval (LSR) and know how to apply it in practice. Specifically, attendees will learn to:

- Compare LSR methods to other transformer-powered ranking methods (i.e., dense retrieval and reranking with cross-encoders)
- Evaluate LSR methods using standard datasets and evaluation practices
- Describe the components of LSR methods, such as the choice of sparse encoders and sparse regularizers, and their associated design decisions
- Compare the impact of LSR methods' components in terms of effectiveness and efficiency
- Describe the design of state-of-the-art LSR methods
- Apply LSR to scenarios outside of monolingual text retrieval, such as multilingual and multimodal settings
- Describe how learned sparse representations can be stored in an inverted index for first-stage retrieval
- Describe strategies for improving the efficiency of retrieval from an inverted index
- Understand how hybrid approaches that further improve effectiveness can be created by combining LSR methods with dense retrieval methods

1.3 Topics and Schedule

The tutorial is planned for a half-day, consisting of two 90-minute sessions with a break in between. The tutorial will be held in person. We propose to cover the following topics:

Part 1 (90 minutes) – Fundamentals

Introduction to LSR [20 min] To introduce LSR, we start with a definition of sparse retrieval. Sparse retrieval is characterized by representing queries and documents as a set of their terms (BoW) [49] and using probability-based methods [13, 14] that score term matches independently from each other [50], with one famous example being BM25 [48]. The main advantage of this natural representation of documents is that it is effective at retrieving documents that have lexical matches to the query, which is a signal of relevance [52]. However, while lexical matching is a useful prior, it can be too restrictive, as it does not include synonyms, regionalisms and other different ways of expressing the same idea [23]. In an effort to fix this lexical mismatch, document and/or query expansion methods [15, 20, 46] have been proposed. Inspired by prior efforts to design both the scoring function and term expansion, LSR methods [22, 35, 61, 64] appeared as a way of learning scoring and expansion rather than defining heuristics.

Datasets and Evaluation [10 min] LSR approaches have most often been trained on the MSMARCO passage retrieval dataset [1], which consists of Web queries and passages. Evaluation is then performed either in-domain or out-of-domain. In-domain can either be the MSMARCO query devset (small 6980 queries version) with the main metric being MRR@10 [58] or the TREC-DL tasks [8–12] that use the MSMARCO corpus, but add new queries and relevance judgments, using nDCG@10 [26] as the main metric. For out-of-domain, the main experiments use the BEIR [57] benchmark, which is where LSR methods showcase their effectiveness, for example by greatly out-performing both BM25 and the dense retrieval methods of the time in [21].

LSR Framework [30 min] Learned sparse retrieval can be described as a framework [45] consisting of three primary components: a *sparse encoder*, a *sparse regularizer*, and a *supervision signal*. The sparse encoder projects raw queries and documents into the vocabulary space, generating lexical or bag-of-words representations. This encoder has two key properties: weighting and expansion. Weighting can either be learned or fixed (e.g., BM25 provides an example of fixed weighting), while expansion can be learned, reused from existing methods, or omitted altogether (with BM25 serving as an example of no expansion). The *sparse regularizer* controls the sparsity of the lexical representation by approximating it through a differentiable weighted loss term. This component is crucial as it affects the size of the inverted index and the retrieval latency of the LSR system. Finally, the *supervision signal* refers to the techniques employed to train the LSR model, such as hard negative mining and distillation from a teacher model.

Text LSR [30 min] The previously introduced framework offers a foundation for analyzing various LSR methods from the literature. In the context of text retrieval, we will examine several prominent approaches, including SNRM [61], DeepImpact [38], EPIC [35], SPLADE [22], and UniCoil [30], through the unified lens of this framework. SNRM [61] was among the first to explore the use of neural networks (specifically, an LSTM network) for learning term expansion and weighting from training data; however, its effectiveness remains limited. The rise of pretrained transformer models, such as DistilBERT [51] and BERT [17], has significantly advanced the field of neural IR [31], and LSR in particular.

DeepImpact [38] leverages a BERT model to re-weight term scores in both queries and documents. EPIC [35] enhances document semantics by extracting expansion terms from the logit matrix of a Masked Language Model (MLM), while keeping queries unexpanded. Similarly, UniCoil [30] shares traits with EPIC but performs document expansion using an external model, Doc2Query [46]. SPLADE [22] goes further by incorporating term weighting and expansion for both queries and documents, utilizing the MLM head. By employing distillation during training, SPLADE [21, 22] achieves competitive performance on MSMARCO and demonstrates strong zero-shot generalization on the BEIR benchmark, surpassing many dense retrieval models.

In this section, alongside discussing the differences between these methods, we will provide a quantitative analysis of how these distinctions, as defined by the framework's components, impact the effectiveness and efficiency of LSR systems.

Part 2 (90 minutes) – Emerging Topics

Multilingual LSR [20 min] While common LSR methods like SPLADE produce representations grounded in an English vocabulary, other work has explored challenging multilingual settings in which the input text and the representation vocabulary are not restricted to a single language [42, 43]. In this setting, using a large multilingual vocabulary can substantially increase computational costs and introduce difficulties aligning representations across languages [42].

Multimodal LSR [20 min] Learned sparse representations are typically grounded in an English vocabulary that is tied to the underlying transformer model (e.g., a WordPiece vocabulary is used with BERT). This property means that representations are transparent, but it creates challenges in settings where the input data is not aligned with the vocabulary. In a multimodal setting, the queries or documents being ranked may not even contain text, such as when matching captions with images. A variety of approaches have been proposed to overcome the challenge of aligning images with textual representations, including distillation-based approaches that adapt existing dense retrieval methods and large-scale approaches that couple a large amount of training data with complex new architectures [6, 34, 44, 62].

Indexing & Efficient LSR [30 min] Inverted Indexes are natural candidates to be used to index and retrieve learned sparse representations, given the extensive research of these data structures in Information Retrieval [18, 19, 32, 36, 39, 40]. Yet, maybe surprisingly, Inverted Indexes do not allow for efficient retrieval when applied to sparse embeddings. The reason is that LSR embeddings fail to comply with some crucial assumptions [2, 4], namely 1) term frequencies following a Zipfian distribution and 2) queries being short. For this purpose, it is common practice to trade the exactness of the search for significant efficiency gains. On the one hand, graph-based solutions meant for searching over dense embeddings have been adapted to work in the sparse domain.[1] Notably, these solutions won the dedicated track at the BigANN@NeurIPS competition in 2023. On the other hand, approximated Inverted Indexes, specially tailored to work with sparse embeddings, have also been proposed [3, 41]. Among them, Seismic [4, 5] excels in efficiency/effectiveness trade-off w.r.t to the other solutions. Example code using Seismic will be provided as part of the support material.

Hybrid Dense-Sparse Retrieval [20 min] Beyond relying solely on learned sparse retrieval for document retrieval, recent works demonstrate that sparse and dense retrieval methods capture complementary relevance signals, such as lexical matching and semantic matching. In this section, we will review recent works that explore the fusion of dense and sparse representations in order to further improve effectiveness [7, 33, 53, 60, 63].

Acknowledgements

This research was supported by project VI.Vidi.223.166 of the NWO Talent Programme which is (partly) financed by the Dutch Research Council (NWO).

References

[1] 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. 2018. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. *arXiv preprint arXiv:1611.09268* (2018).

[2] Sebastian Bruch, Franco Maria Nardini, Amir Ingber, and Edo Liberty. 2023. An approximate algorithm for maximum inner product search over streaming sparse vectors. *ACM Transactions on Information Systems* 42, 2 (2023), 1–43.

[3] Sebastian Bruch, Franco Maria Nardini, Amir Ingber, and Edo Liberty. 2024. Bridging dense and sparse maximum inner product search. *ACM Transactions on Information Systems* (2024).

[4] Sebastian Bruch, Franco Maria Nardini, Cosimo Rulli, and Rossano Venturini. 2024. Efficient Inverted Indexes for Approximate Retrieval over Learned Sparse Representations. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 152–162.

[5] Sebastian Bruch, Franco Maria Nardini, Cosimo Rulli, and Rossano Venturini. 2024. Pairing Clustered Inverted Indexes with kNN Graphs for Fast Approximate Retrieval over Learned Sparse Representations. *arXiv preprint arXiv:2408.04443* (2024).

[6] Chen Chen, Bowen Zhang, Liangliang Cao, Jiguang Shen, Tom Gunter, Albin Jose, Alexander Toshev, Yantao Zheng, Jonathon Shlens, Ruoming Pang, and Yinfei Yang. 2023. STAIR: Learning Sparse Text and Image Representation in Grounded Tokens. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Singapore, 15079–15094.

[7] Xilun Chen, Kushal Lakhotia, Barlas Oguz, Anchit Gupta, Patrick Lewis, Stan Peshterliev, Yashar Mehdad, Sonal Gupta, and Wen-tau Yih. 2022. Salient Phrase Aware Dense Retrieval: Can a Dense Retriever Imitate a Sparse One?. In *Findings of the Association for Computational Linguistics: EMNLP 2022*. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, 250–262.

[8] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, Jimmy Lin, Ellen M. Voorhees, and Ian Soboroff. 2022. Overview of the TREC 2022 Deep Learning Track. In *TREC*.

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

[10] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2021. Overview of the TREC 2020 deep learning track. *arXiv preprint arXiv:2102.07662* (2021). https://arxiv.org/abs/2102.07662

[11] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Hossein A. Rahmani, Daniel Campos, Jimmy Lin, Ellen M. Voorhees, and Ian Soboroff. 2023. Overview of the TREC 2023 Deep Learning Track. In *TREC*.

[12] Nick Craswell1, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Jimmy Lin. 2021. Overview of the TREC 2021 Deep Learning Track. In *TREC*.

[13] Fabio Crestani, Mounia Lalmas, Cornelis J. Van Rijsbergen, and Iain Campbell. 1998. "Is this document relevant?...probably": a survey of probabilistic models in information retrieval. *ACM Comput. Surv.* 30, 4 (1998), 528–552.

[14] W. Bruce Croft. 1981. Document representation in probabilistic models of information retrieval. *Journal of the American Society for Information Science* 32, 6 (1981), 451–457.

[15] W. Bruce Croft and David J. Harper. 1979. Probabilistic models of document retrieval with relevance information. *Journal of Documentation* 35, 4 (1979), 285–295.

[16] Zhuyun Dai and Jamie Callan. 2020. Context-Aware Document Term Weighting for Ad-Hoc Search. In *Proceedings of The Web Conference 2020* (Taipei, Taiwan) *(WWW '20)*. Association for Computing Machinery, 1897–1907.

[17] Jacob Devlin. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[18] Constantinos Dimopoulos, Sergey Nepomnyachiy, and Torsten Suel. 2013. Optimizing top-k document retrieval strategies for block-max indexes. In *Proceedings of the sixth ACM international conference on Web search and data mining*. 113–122.

[19] Shuai Ding and Torsten Suel. 2011. Faster top-k document retrieval using block-max indexes. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*. 993–1002.

[20] Miles Efron, Peter Organisciak, and Katrina Fenlon. 2012. Improving retrieval of short texts through document expansion *(SIGIR '12)*. Association for Computing Machinery, 911–920.

[21] Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. 2021. SPLADE v2: Sparse Lexical and Expansion Model for Information Retrieval. *arXiv preprint arXiv:2109.10086* (2021).

[22] 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, 2288–2292.

[23] G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais. 1987. The vocabulary problem in human-system communication. *Commun. ACM* 30, 11 (1987), 964–971.

[24] Zhichao Geng, Xinyuan Lu, Dagney Braun, Charlie Yang, and Fanit Kolchina. 2023. Improving document retrieval with sparse semantic encoders. https://opensearch.org/blog/improving-document-retrieval-with-sparse-semantic-encoders/

[25] Seraphina Goldfarb-Tarrant, Pedro Rodriguez, Jane Dwivedi-Yu, and Patrick Lewis. 2024. MultiContrievers: Analysis of Dense Retrieval Representations. *arXiv preprint arXiv:2402.15925* (2024).

[26] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst.* 20, 4 (2002), 422–446.

[27] 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)*. 6769–6781.

[28] Carlos Lassance and Stéphane Clinchant. 2022. An Efficiency Study for SPLADE Models. In *Proceedings of the 45th International ACM SIGIR Conference on Research*

[1] https://github.com/Leslie-Chung/GrassRMA, https://github.com/veaaaab/pyanns

and Development in Information Retrieval (Madrid, Spain) *(SIGIR '22)*. Association for Computing Machinery, 2220–2226.

[29] Xiaoxi Li, Jiajie Jin, Yujia Zhou, Yuyao Zhang, Peitian Zhang, Yutao Zhu, and Zhicheng Dou. 2024. From matching to generation: A survey on generative information retrieval. *arXiv preprint arXiv:2404.14851* (2024).

[30] Jimmy Lin and Xueguang Ma. 2021. A Few Brief Notes on DeepImpact, COIL, and a Conceptual Framework for Information Retrieval Techniques. *arXiv preprint arXiv:2106.14807* (2021).

[31] Jimmy Lin, Rodrigo Nogueira, and Andrew Yates. 2020. Pretrained Transformers for Text Ranking: BERT and Beyond. *ArXiv* abs/2010.06467 (2020). https://arxiv.org/abs/2010.06467

[32] Jimmy Lin and Andrew Trotman. 2015. Anytime ranking for impact-ordered indexes. In *Proceedings of the 2015 International Conference on The Theory of Information Retrieval*. 301–304.

[33] Sheng-Chieh Lin and Jimmy Lin. 2023. A Dense Representation Framework for Lexical and Semantic Matching. *ACM Trans. Inf. Syst.* (2023).

[34] Ziyang Luo, Pu Zhao, Can Xu, Xiubo Geng, Tao Shen, Chongyang Tao, Jing Ma, Qingwei Lin, and Daxin Jiang. 2023. LexLIP: Lexicon-Bottlenecked Language-Image Pre-Training for Large-Scale Image-Text Sparse Retrieval. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 11206–11217.

[35] 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* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, 1573–1576.

[36] Joel Mackenzie, Matthias Petri, and Luke Gallagher. 2022. IOQP: A simple Impact-Ordered Query Processor written in Rust. In *Proceedings of the Third International Conference on Design of Experimental Search & Information REtrieval Systems, San Jose, CA, USA, August 30-31, 2022 (CEUR Workshop Proceedings, Vol. 3480)*, Omar Alonso, Ricardo Baeza-Yates, Tracy Holloway King, and Gianmaria Silvello (Eds.). CEUR-WS.org, 22–34. https://ceur-ws.org/Vol-3480/paper-03.pdf

[37] Yury Malkov, Alexander Ponomarenko, Andrey Logvinov, and Vladimir Krylov. 2014. Approximate nearest neighbor algorithm based on navigable small world graphs. *Inf. Syst.* 45 (2014), 61–68. https://doi.org/10.1016/J.IS.2013.10.006

[38] 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, 1723–1727.

[39] Antonio Mallia, Giuseppe Ottaviano, Elia Porciani, Nicola Tonellotto, and Rossano Venturini. 2017. Faster BlockMax WAND with variable-sized blocks. In *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval*. 625–634.

[40] Antonio Mallia and Elia Porciani. 2019. Faster BlockMax WAND with longer skipping. In *Advances in Information Retrieval: 41st European Conference on IR Research, ECIR 2019, Cologne, Germany, April 14–18, 2019, Proceedings, Part I 41*. Springer, 771–778.

[41] Antonio Mallia, Torsten Suel, and Nicola Tonellotto. 2024. Faster learned sparse retrieval with block-max pruning. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 2411–2415.

[42] Suraj Nair, Eugene Yang, Dawn Lawrie, James Mayfield, and Douglas W. Oard. 2023. BLADE: Combining Vocabulary Pruning and Intermediate Pretraining for Scaleable Neural CLIR. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Taipei, Taiwan) *(SIGIR '23)*. Association for Computing Machinery, 1219–1229.

[43] Suraj Nair, Eugene Yang, Dawn J Lawrie, James Mayfield, and Douglas W. Oard. 2022. Learning a Sparse Representation Model for Neural CLIR. In *Biennial Conference on Design of Experimental Search & Information Retrieval Systems*.

[44] Thong Nguyen, Mariya Hendriksen, Andrew Yates, and Maarten de Rijke. 2024. Multimodal Learned Sparse Retrieval with Probabilistic Expansion Control. In *Advances in Information Retrieval: 46th European Conference on Information Retrieval, ECIR 2024, Glasgow, UK, March 24–28, 2024, Proceedings, Part II* (Glasgow, United Kingdom). Springer-Verlag, 448–464.

[45] Thong Nguyen, Sean MacAvaney, and Andrew Yates. 2023. A Unified Framework for Learned Sparse Retrieval. In *Advances in Information Retrieval: 45th European Conference on Information Retrieval, ECIR 2023, Dublin, Ireland, April 2–6, 2023, Proceedings, Part III* (Dublin, Ireland). Springer-Verlag, 101–116.

[46] Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019. Document Expansion by Query Prediction. *arXiv preprint arXiv:1904.08375* (2019).

[47] Ronak Pradeep, Kai Hui, Jai Gupta, Adam Lelkes, Honglei Zhuang, Jimmy Lin, Donald Metzler, and Vinh Tran. 2023. How Does Generative Retrieval Scale to Millions of Passages?. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 1305–1321. https://doi.org/10.18653/v1/2023.emnlp-main.83

[48] Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at TREC-3. *Nist Special Publication Sp* 109 (1995), 109.

[49] Gerard Salton and Chris Buckley. 1988. Term-Weighting Approaches in Automatic Text Retrieval. *Inf. Process. Manag.* 24 (1988), 513–523.

[50] G. Salton, A. Wong, and C. S. Yang. 1975. A vector space model for automatic indexing. *Commun. ACM* 18, 11 (1975), 613–620.

[51] V Sanh. 2019. DistilBERT, A Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter. *arXiv preprint arXiv:1910.01108* (2019).

[52] Tefko Saracevic. 2016. The Notion of Relevance in Information Science. In *Morgan & Claypool Publishers*.

[53] Tao Shen, Xiubo Geng, Chongyang Tao, Can Xu, Guodong Long, Kai Zhang, and Daxin Jiang. 2023. UnifieR: A Unified Retriever for Large-Scale Retrieval. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA) *(KDD '23)*. Association for Computing Machinery, 4787–4799.

[54] Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is ChatGPT Good at Search? Investigating Large Language Models as Re-Ranking Agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Computational Linguistics, Singapore, 14918–14937. https://doi.org/10.18653/v1/2023.emnlp-main.923

[55] Yubao Tang, Ruqing Zhang, Jiafeng Guo, and Maarten de Rijke. 2023. Recent Advances in Generative Information Retrieval. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region, SIGIR-AP 2023, Beijing, China, November 26-28, 2023*, Qingyao Ai, Yiqin Liu, Alistair Moffat, Xuanjing Huang, Tetsuya Sakai, and Justin Zobel (Eds.). ACM, 294–297. https://doi.org/10.1145/3624918.3629547

[56] Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. 2022. Transformer memory as a differentiable search index. *Advances in Neural Information Processing Systems* 35 (2022), 21831–21843.

[57] Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. *arXiv preprint arXiv:2104.08663* (2021).

[58] Ellen M. Voorhees and Dawn M. Tice. 1999. The TREC-8 Question Answering Track Report. In *Proceedings of the Eighth Text REtrieval Conference (TREC-8)*.

[59] Ikuya Yamada, Akari Asai, and Hannaneh Hajishirzi. 2021. Efficient Passage Retrieval with Hashing for Open-domain Question Answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (Eds.). Association for Computational Linguistics, Online, 979–986.

[60] Yingrui Yang, Parker Carlson, Shanxiu He, Yifan Qiao, and Tao Yang. 2024. Cluster-based Partial Dense Retrieval Fused with Sparse 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, 2327–2331.

[61] 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, 497–506.

[62] Jiawei Zhou, Xiaoguang Li, Lifeng Shang, Xin Jiang, Qun Liu, and Lei Chen. 2024. Retrieval-based Disentangled Representation Learning with Natural Language Supervision. In *The Twelfth International Conference on Learning Representations*.

[63] Shengyao Zhuang, Xueguang Ma, Bevan Koopman, Jimmy Lin, and Guido Zuccon. 2024. PromptReps: Prompting Large Language Models to Generate Dense and Sparse Representations for Zero-Shot Document Retrieval. *arXiv preprint arXiv:2404.18424* (2024).

[64] Shengyao Zhuang and Guido Zuccon. 2021. Fast Passage Re-ranking with Contextualized Exact Term Matching and Efficient Passage Expansion. *arXiv preprint arXiv:2108.08513* (2021).