Learning Discrete Representations via Constrained Clustering for Effective and Efficient Dense Retrieval (Extended Abstract)[*]

Jingtao Zhan[1] , **Jiaxin Mao**[2] , **Yiqun Liu**[1†] , **Jiafeng Guo**[3] , **Min Zhang**[1] and **Shaoping Ma**[1]
[1]Department of Computer Science and Technology, Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing 100084, China
[2]Beijing Key Laboratory of Big Data Management and Analysis Methods, Gaoling School of Artificial Intelligence, Renmin University of China, Beijing 100872, China
[3]CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China
yiqunliu@tsinghua.edu.cn

Abstract

Dense Retrieval (DR) has achieved state-of-the-art first-stage ranking effectiveness. However, the efficiency of most existing DR models is limited by the large memory cost of storing dense vectors and the time-consuming nearest neighbor search (NNS) in vector space. Therefore, we present RepCONC, a novel retrieval model that learns discrete Representations via CONstrained Clustering. RepCONC jointly trains dual-encoders and the Product Quantization (PQ) method to learn discrete document representations and enables fast approximate NNS with compact indexes. It models quantization as a constrained clustering process, which requires the document embeddings to be uniformly clustered around the quantization centroids. We theoretically demonstrate that the uniform clustering constraint facilitates representation distinguishability. Extensive experiments show that RepCONC substantially outperforms a wide range of existing retrieval models in terms of retrieval effectiveness, memory efficiency, and time efficiency.

1 Introduction

Dense Retrieval (DR) has become a popular paradigm for first-stage retrieval in ad-hoc retrieval tasks. It embeds queries and documents in a latent vector space with dual-encoders and uses nearest neighbor search to retrieve relevant documents. With end-to-end supervised training, DR models have achieved state-of-the-art ranking performance and significantly outperform BoW models [Zhan *et al.*, 2021b; Lin *et al.*, 2020; Xiong *et al.*, 2021].

Despite the success in improving ranking performance, most existing DR models [Zhan *et al.*, 2021b; Xiong *et al.*, 2021; Karpukhin *et al.*, 2020] are inefficient in memory usage and retrieval speed. For memory inefficiency, the size of the embedding index is usually an order of magnitude larger than that of BoW index [Zhan *et al.*, 2021a]. At runtime, the vectors must be loaded to costly system memory or even GPU memory. As for time inefficiency, many existing DR models conduct exhaustive search, i.e., computing relevance scores between the submitted query and all documents. As a result, these DR models cannot use CPUs for retrieval due to high latency and have to use much more expensive GPUs to accelerate the search.

To tackle this problem, we present RepCONC, which stands for learning discrete **Rep**resentations via **CON**strained **C**lustering[1]. RepCONC learns discrete representations with Product Quantization (PQ) so that the representations can be encoded into compact indexes for efficient vector search. RepCONC utilizes joint optimization of dual-encoders and PQ to achieve effective ranking results. During joint optimization, RepCONC models quantization as a *constrained clustering* process, which involves a clustering loss and a uniform clustering constraint. The clustering loss is introduced to train the discrete codes. And the uniform clustering constraint facilitates distinguishability of discrete representations by requiring the vectors to be equally assigned to all quantization centroids. Besides *constrained clustering*, RepCONC further employs vector-based inverted file system (IVF) [Jegou *et al.*, 2010] to enable efficient non-exhaustive vector search on either GPU or CPU.

We conduct experiments on two widely-adopted ad-hoc retrieval benchmarks [Bajaj *et al.*, 2016; Craswell *et al.*, 2020] and compare RepCONC with a wide range of baselines. Experimental results show that: 1) RepCONC significantly outperforms competitive vector compression baselines with different compression ratio settings. 2) RepCONC substantially outperforms various retrieval baselines in terms of retrieval effectiveness, memory efficiency, and time efficiency.

2 Constrained Clustering Model

In this section, we propose RepCONC, which stands for learning discrete **Rep**resentations via **CON**strained

[*] This is an abridged version of the paper [Zhan *et al.*, 2022] that won the Best Paper Award at ACM WSDM 2022.

[†] Corresponding Author

[1] Code and models are available at https://github.com/jingtaozhan/RepCONC.

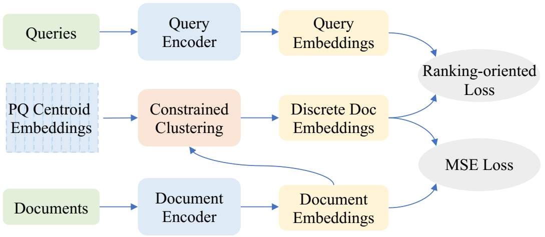

Figure 1: Training process of RepCONC.

Clustering. We first introduce the preliminary of Production Quantization [Jegou *et al.*, 2010] and then elaborate on our model.

2.1 Revisiting Product Quantization

RepCONC is based on Product Quantization (PQ) [Jegou *et al.*, 2010]. For vectors of dimension D, PQ defines M sets of embeddings, each of which includes K embeddings of dimension D/M. They are called PQ Centroid Embeddings. Formally, let $\boldsymbol{c}_{i,j}$ be the j_{th} centroid embedding from the i_{th} set:

$$\boldsymbol{c}_{i,j} \in \mathbb{R}^{\frac{D}{M}} \quad (1 \le i \le M, 1 \le j \le K) \tag{1}$$

Given a document embedding $\boldsymbol{d} \in \mathbb{R}^D$, PQ firstly splits it into M sub-vectors.

$$\boldsymbol{d} = \boldsymbol{d}_1, \boldsymbol{d}_2, ..., \boldsymbol{d}_M \tag{2}$$

Then PQ independently quantizes each sub-vector to the nearest PQ Centroid Embedding. Formally, to quantize a sub-vector $\boldsymbol{d}_i$, PQ selects the nearest $\boldsymbol{c}_{i,\varphi_i(d)}$:

$$\varphi_i(d) = \arg\min_j \|\boldsymbol{c}_{i,j} - \boldsymbol{d}_i\|^2 \tag{3}$$

Let $\boldsymbol{\varphi}(d)$ be the concatenation of $\varphi_i(d)$:

$$\boldsymbol{\varphi}(d) = \varphi_1(d), \varphi_2(d), ..., \varphi_i(M) \in \{1, 2, ..., K\}^M \tag{4}$$

where comma denotes vector concatenation. $\boldsymbol{\varphi}(d)$ is called the Index Assignment of d. $\boldsymbol{\varphi}(d)$ can reconstruct the quantized document embedding $\hat{\boldsymbol{d}}$ as follows:

$$\hat{\boldsymbol{d}} = \boldsymbol{c}_{1,\varphi_1(d)}, \boldsymbol{c}_{2,\varphi_2(d)}, ..., \boldsymbol{c}_{M,\varphi_M(d)} \in \mathbb{R}^D \tag{5}$$

PQ does not explicitly store $\boldsymbol{d}$ or $\hat{\boldsymbol{d}}$. It only stores the PQ Centroid Embeddings $\{\boldsymbol{c}_{i,j}\}$ and Index Assignments $\boldsymbol{\varphi}(d)$.

2.2 Clustering and Representation Learning

RepCONC views joint optimization as a simultaneous clustering and representation learning problem. It utilizes both the ranking-oriented loss [Zhan *et al.*, 2021a] and a clustering loss. We illustrate the training workflow in Figure 1.

The ranking-oriented loss computes the ranking loss based on the compressed document embeddings. Therefore, it better evaluates the ranking performance with respect to the current compression parameters. Let d^+ and d^- be relevant and irrelevant documents, respectively. Ranking-oriented loss L_r is formulated as:

$$L_r = -\log \frac{\mathrm{e}^{\langle \boldsymbol{q}, \hat{\boldsymbol{d}}^+ \rangle}}{\mathrm{e}^{\langle \boldsymbol{q}, \hat{\boldsymbol{d}}^+ \rangle} + \sum_{d^-} \mathrm{e}^{\langle \boldsymbol{q}, \hat{\boldsymbol{d}}^- \rangle}} \tag{6}$$

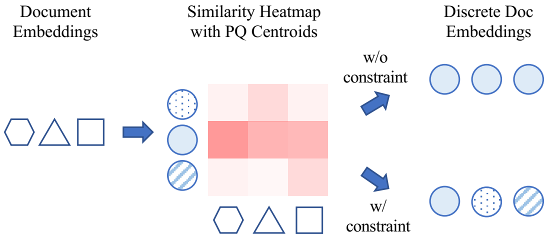

Figure 2: Illustration of Constrained Clustering. Darker colors in the heatmap indicate higher similarities (smaller distances). With the constraint, the discrete document embeddings are more diverse.

RepCONC regards quantization as a clustering problem and additionally introduces the MSE loss L_m:

$$L_m = \|\boldsymbol{d} - \hat{\boldsymbol{d}}\|^2 \tag{7}$$

Minimizing L_m requires the document embeddings to be clustered around the centroid embeddings. Combining both L_r and L_m helps the model to cluster document embeddings based on ranking effectiveness. It is expected to produce better clustering compared with unsupervised training. The final loss L is a weighted sum of ranking-oriented loss L_r and the MSE loss L_m.

$$L = L_r + \lambda L_m \tag{8}$$

Since quantization is non-differentiable, we explicitly design the gradient back-propagation policy for document encoders. The gradients of uncompressed document embeddings are defined as follows:

$$\frac{\partial L}{\partial \boldsymbol{d}} := \frac{\partial L_r}{\partial \hat{\boldsymbol{d}}} + \lambda \frac{\partial L_m}{\partial \boldsymbol{d}} \tag{9}$$

As the equation shows, we add the gradient of quantized document embeddings (the first term). The gradients are further back-propagated to document encoders. Gradients of other parameters can be derived with chain rule.

2.3 Importance of Uniform Clustering

It is non-trivial to simultaneously conduct clustering and representation learning because the two objects are conflicting to some extent. Although representation learning encourages vectors to be distinguishable, clustering encourages vectors to be identical. In the iterative training process, clustering objective leads to unbalanced clustering distribution, which affects the vector distinguishability and compromises ranking effectiveness.

We tackle this challenge by imposing a uniform clustering constraint. It requires the document sub-vectors to be equally assigned to all PQ Centroid Embeddings. The learning object along with the constraint is formally expressed as:

$$\min L \quad \text{subject to } \forall i, j : P(\varphi_i(d) = j) = \frac{1}{K} \tag{10}$$

We illustrate constrained clustering in Figure 2. As the figure shows, the discrete document embeddings are selected by minimizing the quantization error (maximizing the similarity) given the uniform clustering constraint. Without the constraint, the discrete document embeddings become identical. Now we theoretically analyze the importance of uniform clustering. Due to the limitation of space, we summarize the theoretical conclusions here and refer readers to the full paper [Zhan *et al.*, 2022] for detailed derivation.

Theorem 1. *Maximizing the distinguishability of vectors is equivalent to forcing the vectors to be equally quantized to all possible Index Assignments.*

That is to say, quantizing vectors equally to all possible Index Assignments helps representations to be distinguishable.

Theorem 2. *Uniformly clustering sub-vectors is the essential condition of maximum distinguishability.*

Theorem 3. *If sub-vectors are independent, uniformly clustering sub-vectors is the sufficient condition of maximum distinguishability.*

Although independence among sub-vectors may not hold for practical dual-encoders, we believe uniformly clustering sub-vectors is still helpful for distinguishing quantized vectors.

2.4 Constrained Clustering Optimization

This section shows how to incorporate the uniform clustering constraint to select Index Assignments during training.

We introduce a posterior distribution $q(j|\boldsymbol{d}_i)$, which is the probability that the sub-vector $\boldsymbol{d}_i$ is quantized to the centroid $\boldsymbol{c}_{i,j}$. The Index Assignment, $\varphi_i(d)$, is the centroid with the maximum probability:

$$\varphi_i(d) = \arg\max_j q(j|\boldsymbol{d}_i) \tag{11}$$

For PQ that uses Eq. (3), $q(j|\boldsymbol{d}_i)$ can be regarded as being computed solely based on quantization error. Here for RepCONC, we compute $q(j|\boldsymbol{d}_i)$ by minimizing the quantization error given the uniform clustering constraint:

$$\forall i: \min_q \sum_{d\in\mathcal{D}} \sum_{j=1}^{K} q(j|\boldsymbol{d}_i)\|\boldsymbol{c}_{i,j} - \boldsymbol{d}_i\|^2 \text{ subject to}$$
$$\forall j, d: q(j|\boldsymbol{d}_i) \in \{0,1\}, \sum_{j=1}^{K} q(j|\boldsymbol{d}_i) = 1, \text{ and } \sum_{d\in\mathcal{D}} q(j|\boldsymbol{d}_i) = \frac{|\mathcal{D}|}{K} \tag{12}$$

where $\mathcal{D}$ indicates the set of all documents. The first condition constrains $q(j|\boldsymbol{d}_i)$ to be binary, the second condition is a natural requirement for probability, and the third condition is exactly the uniform clustering constraint. Without the third condition, Eq. (11) and (12) degenerate to Eq. (3), i.e., selecting Index Assignments with minimum quantization error.

Solving Eq. (12) is particularly difficult because it is a combinatorial optimization problem with the scale of millions or even billions of documents. Therefore, we use an approximate solution by relaxing q to be continuous and focusing on uniformly clustering a mini-batch of documents $\mathcal{B}$:

$$\forall i: \min_q \sum_{d\in\mathcal{B}} \sum_{j=1}^{K} q(j|\boldsymbol{d}_i)\|\boldsymbol{c}_{i,j} - \boldsymbol{d}_i\|^2$$
$$\text{subject to } \forall d: \sum_{j=1}^{K} q(j|\boldsymbol{d}_i) = 1 \text{ and } \forall j: \sum_{d\in\mathcal{B}} q(j|\boldsymbol{d}_i) = \frac{|\mathcal{B}|}{K} \tag{13}$$

Since $\|\boldsymbol{c}_{i,j} - \boldsymbol{d}_i\|^2$ can be regarded as the cost of mapping $\boldsymbol{d}_i$ to $\boldsymbol{c}_{i,j}$, this is an instance of the optimal transport problem and can be solved in polynomial time by linear program. In our implementation, we use Sinkhorn-Knopp algorithm [Cuturi, 2013] to efficiently solve Eq. (13).

2.5 Accelerating Search with IVF

Besides PQ, RepCONC employs the inverted file system (IVF) to accelerate vector search. After quantizing document embeddings, RepCONC uses k-means to generate n clusters. Given a query embedding, RepCONC selects the nearest $\tilde{n}$ clusters and only ranks the documents in them. The documents in other clusters are ignored. In this way, RepCONC approximately accelerates vector search by $n/\tilde{n}$.

3 Experimental Setup

Here we present our experimental settings.

3.1 Datasets and Metrics

We conduct experiments on two large-scale ad-hoc retrieval benchmarks from the TREC 2019 Deep Learning Track [Craswell *et al.*, 2020; Bajaj *et al.*, 2016], passage ranking and document ranking. Due to the limited space, this paper only reports the performance on the passage ranking task. Please refer to our full paper [Zhan *et al.*, 2022] for comprehensive results. The passage ranking task has a corpus of 8.8M passages, $0.5M$ training queries, $7k$ development queries (henceforth, MARCO Passage), and 43 test queries (DL Passage). We report the official metrics and R@100 based on the full-corpus retrieval results.

3.2 Baselines

We exploit two types of baselines, vector compression methods and retrieval models.

For vector compression methods, we adopt both unsupervised and supervised methods. The former include PQ [Jegou *et al.*, 2010], ScaNN [Guo *et al.*, 2020], ITQ+LSH [Gong *et al.*, 2012], OPQ [Ge *et al.*, 2013], and OPQ+ScaNN. The latter include DPQ [Chen *et al.*, 2020] and JPQ [Zhan *et al.*, 2021a].

For retrieval baselines, we utilize BoW models, dual-encoders, and some competitive complex retrieval systems. BoW models involve BM25 [Robertson and Walker, 1994], DeepCT [Dai and Callan, 2019], doc2query [Nogueira *et al.*, 2019b], and docT5query [Nogueira *et al.*, 2019a]. Dual-encoders include RepBERT [Zhan *et al.*, 2020], ANCE [Xiong *et al.*, 2021], and ADORE [Zhan *et al.*, 2021b]. Complex retrieval systems are much slower than BoW and dual-encoders. They include ColBERT [Khattab and Zaharia, 2020], COIL [Gao *et al.*, 2021], uniCOIL [Lin and Ma, 2021], and DeepImpact [Mallia *et al.*, 2021].

4 Experiments

We empirically evaluate RepCONC in this section.

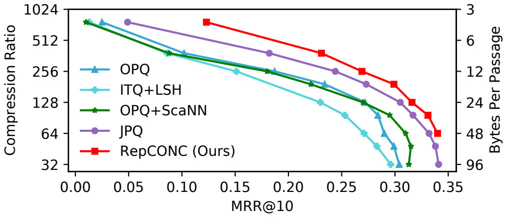

Figure 3: Comparison with compression methods. Up and right is better.

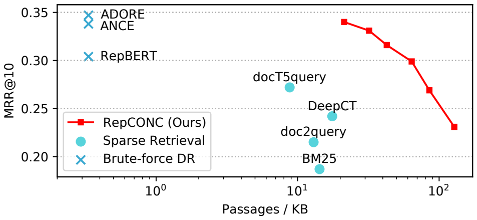

Figure 4: Comparison with first-stage retrieval models in terms of effectiveness-memory trade-off. Up and right is better. The x-axis indicates the average number of passages/documents stored in 1 kilobyte.

4.1 Comparison with Compression Methods

This section compares RepCONC with vector compression baselines to answer **RQ1**. Ranking performances in terms of different compression ratios are plotted in Figure 3. The advantage of RepCONC is more significant when larger compression ratios are used. For example, its MRR score is more than twice the JPQ's score when the compression ratio is 784x. We believe this is because RepCONC is able to generate high-quality Index Assignments specifically for ranking effectiveness, which becomes more important when fewer bytes are used. Instead, JPQ uses K-Means to produce task-blind Index Assignments and compromises ranking performance.

4.2 Comparison with Retrieval Models

This section compares RepCONC with various retrieval models to address **RQ2**. We firstly compare it with efficient first-stage retrievers and then compare it with complex (slow) end-to-end retrievers.

Comparison with First-Stage Retrievers

Figure 4 summarizes the effectiveness-memory tradeoff. As the figure shows, although DR models are much more effective than BoW models, they incur severe memory inefficiency. By jointly training the dual-encoders and quantization methods, RepCONC substantially improves memory efficiency of DR while still being very effective in ranking. It outperforms RepBERT [Zhan *et al.*, 2020] and ANCE [Xiong *et al.*, 2021] in effectiveness, and is almost as effective as ADORE [Zhan *et al.*, 2021b], the state-of-the-art DR model trained by negative sampling.

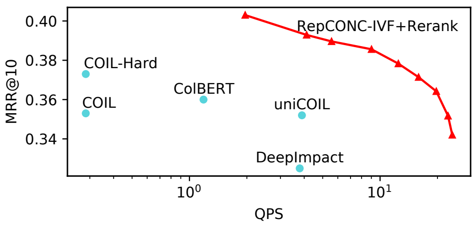

Figure 5: Comparison with complex (slow) end-to-end retrieval models in terms of effectiveness-latency tradeoff on MS MARCO Passage Ranking. The search is performed on CPU with one thread. Up and right is better. QPS stands for 'query per second'.

Comparison with Complex End-to-End Retrievers

This section compares RepCONC with some complex (slow) end-to-end neural retrieval models. These models achieve better ranking performance with much higher query latency because of their complex model architecture. In consideration of fair comparison, we add a reranking stage to RepCONC and compare them in terms of effectiveness-latency tradeoff. The reranking models are MonoBERT and DuoT5 models open-sourced by the pygaggle library [2]. Note, query encoding and reranking are performed on GPU while the search is performed on CPU with one thread. Ranking results are shown in Figure 5. We can see that RepCONC-IVF+Rerank substantially outperforms all baselines in terms of both effectiveness and time efficiency.

5 Conclusions

To solve the efficiency issue existing in brute-force DR models, we present RepCONC, which learns discrete representations by modeling quantization as constrained clustering in the joint learning process. The clustering object requires the document embeddings to be clustered around the quantization centroids and facilitates joint optimization of PQ parameters and dual-encoders. We also introduce a uniform clustering constraint to maximize the representation distinguishability. We conduct experiments on widely-adopted ad-hoc retrieval benchmarks. Experimental results show that RepCONC significantly outperforms competitive quantization baselines and substantially improves the memory efficiency and time efficiency of DR.

Acknowledgments

This work is supported by the Natural Science Foundation of China (Grant No. 61732008) and Tsinghua University Guoqiang Research Institute.

References

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

[2] https://github.com/castorini/pygaggle

[Chen *et al.*, 2020] Ting Chen, Lala Li, and Yizhou Sun. Differentiable product quantization for end-to-end embedding compression. In *International Conference on Machine Learning*, pages 1617–1626. PMLR, 2020.

[Craswell *et al.*, 2020] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. Overview of the trec 2019 deep learning track. In *Text REtrieval Conference (TREC)*. TREC, 2020.

[Cuturi, 2013] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26:2292–2300, 2013.

[Dai and Callan, 2019] Zhuyun Dai and Jamie Callan. Context-aware sentence/passage term importance estimation for first stage retrieval. *arXiv preprint arXiv:1910.10687*, 2019.

[Gao *et al.*, 2021] Luyu Gao, Zhuyun Dai, and Jamie Callan. COIL: Revisit exact lexical match in information retrieval with contextualized inverted list. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3030–3042, Online, June 2021.

[Ge *et al.*, 2013] Tiezheng Ge, Kaiming He, Qifa Ke, and Jian Sun. Optimized product quantization. *IEEE transactions on pattern analysis and machine intelligence*, 36(4):744–755, 2013.

[Gong *et al.*, 2012] Yunchao Gong, Svetlana Lazebnik, Albert Gordo, and Florent Perronnin. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *IEEE transactions on pattern analysis and machine intelligence*, 35(12):2916–2929, 2012.

[Guo *et al.*, 2020] Ruiqi Guo, Philip Sun, Erik Lindgren, Quan Geng, David Simcha, Felix Chern, and Sanjiv Kumar. Accelerating large-scale inference with anisotropic vector quantization. In *International Conference on Machine Learning*, pages 3887–3896. PMLR, 2020.

[Jegou *et al.*, 2010] Herve Jegou, Matthijs Douze, and Cordelia Schmid. Product quantization for nearest neighbor search. *IEEE transactions on pattern analysis and machine intelligence*, 33(1):117–128, 2010.

[Karpukhin *et al.*, 2020] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2020.

[Khattab and Zaharia, 2020] O. Khattab and M. Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert. *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020.

[Lin and Ma, 2021] Jimmy Lin and Xueguang Ma. A few brief notes on deepimpact, coil, and a conceptual framework for information retrieval techniques. *arXiv preprint arXiv:2106.14807*, 2021.

[Lin *et al.*, 2020] Sheng-Chieh Lin, Jheng-Hong Yang, and Jimmy Lin. Distilling dense representations for ranking using tightly-coupled teachers. *arXiv preprint arXiv:2010.11386*, 2020.

[Mallia *et al.*, 2021] Antonio Mallia, Omar Khattab, Torsten Suel, and Nicola Tonellotto. Learning passage impacts for inverted indexes. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, 2021.

[Nogueira *et al.*, 2019a] Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. From doc2query to docTTTTTquery. *Online preprint*, 2019.

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

[Robertson and Walker, 1994] Stephen E Robertson and Steve Walker. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR'94*, pages 232–241. Springer, 1994.

[Xiong *et al.*, 2021] Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *International Conference on Learning Representations*, 2021.

[Zhan *et al.*, 2020] Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Min Zhang, and Shaoping Ma. Repbert: Contextualized text embeddings for first-stage retrieval. *arXiv preprint arXiv:2006.15498*, 2020.

[Zhan *et al.*, 2021a] Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. Jointly optimizing query encoder and product quantization to improve retrieval performance. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management*, 2021.

[Zhan *et al.*, 2021b] Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. Optimizing dense retrieval model training with hard negatives. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, page 1503–1512, 2021.

[Zhan *et al.*, 2022] Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. Learning discrete representations via constrained clustering for effective and efficient dense retrieval. In *Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining*, WSDM '22, page 1328–1336. Association for Computing Machinery, 2022.