BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Excel@FIT 2025
http://excel.fit.vutbr.cz

Efficient Multi-Vector Document Retrieval with Adaptive Representation Size

Jakub Štětina*

Abstract
This work presents ColBERT-Sparse, an extension of the ColBERTv2 PLAID information retrieval model aimed at reducing index sizes for large-scale search. Sparsity scoring layers are introduced to selectively retain only important token embeddings. Experiments show that significant index compression can be achieved while maintaining high retrieval effectiveness. The results suggest promising directions for making dense retrieval models more scalable and efficient.

*xsteti05@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

Efficient document retrieval remains a challenging problem for multi-vector information retrieval (IR) models as text corpora grow to large scales. State-of-the-art models like ColBERTv2 PLAID [1, 2] offer excellent retrieval accuracy, but they lead to major memory and storage overheads due to the need to index large numbers of token-level embeddings.

This project introduces *ColBERT-Sparse*, an adaptive sparsification method that selectively prunes token-level embeddings during indexing. The goal is to substantially reduce index size while preserving retrieval performance as much as possible.

The ColBERT-based retrieval method [3] stores hundreds of embeddings per document, leading to index sizes that scale poorly with corpus growth. By learning to identify and retain only the most retrieval-relevant tokens, ColBERT-Sparse addresses this bottleneck without modifying the core late interaction retrieval mechanism that the original ColBERT model [3] uses.

Sparse retrieval methods such as SPLADE [4] and RAPTOR [5] tackle similar scalability issues but operate differently: typically at the vocabulary level or through hierarchical structures. In contrast, ColBERT-Sparse focuses on embedding-level sparsification directly within the ColBERT retrieval paradigm.

A sparsity scoring layer is incorporated on top of ColBERT's token embeddings to enhance model efficiency. To encourage sparsity during training, the loss function is modified by adding an L1 penalty term (as shown in Formula 1), which helps guide the model toward sparse representations. The degree of learned sparsification can be further controlled by adjusting the λ parameter in Formula 1. Two distinct variants of the sparsity mechanism are explored: sigmoid thresholding and Gumbel-Softmax selection [6]. The final decision in the sigmoid thresholding variant is based on whether the sparsity value for a given token crosses a predefined threshold, or, in the Gumbel-Softmax variant, the model selects by itself the token to keep or discard by drawing from a categorical distribution formed via a softmax function, allowing for probabilistic token selection.

The ColBERT-Sparse architecture is visualized in Figure 3. The distribution of learned sparsity scores during training (Figure 2) demonstrates how the model gradually shifts from dense to sparse token retention, with the majority of tokens receiving scores near zero.

Apart from the architectural modifications and adjustments to the objective function, the training procedure remained consistent with the original approach from the ColBERTv2 PLAID model, utilizing distillation from a 22M-parameter MiniLM cross-encoder [7]. All experiments were evaluated using a 60k-sample subset of the development set from the MS MARCO dataset [8].

2. Contributions

- Developed a learnable sparsity mechanism for dense retrieval (Figure 3).
- Achieved an average token reduction of >80% while maintaining near-baseline recall@10 (Table 1).
- Analyzed part-of-speech retention patterns, showing semantic alignment in pruning (Figure 4, Table 2).
- Analyzed the impact of sparsification across different document lengths (Figure 4).
- Studied training dynamics and sparsity score evolution across checkpoints (Figure 1 & 2).

3. Conclusions

ColBERT-Sparse demonstrates that adaptive token pruning is a viable strategy for scalable dense retrieval, offering significant reductions in memory and storage requirements with minimal performance loss. Future work will explore jointly optimizing token retention and centroid estimation to improve end-to-end retrieval efficiency.

Acknowledgements

I would like to thank my supervisor for their guidance and support throughout this project.

References

[1] Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. Colbertv2: Effective and efficient retrieval via lightweight late interaction, 2022.

[2] Keshav Santhanam, Omar Khattab, Christopher Potts, and Matei Zaharia. Plaid: An efficient engine for late interaction retrieval, 2022.

[3] Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. Colbertv2: Effective and efficient retrieval via lightweight late interaction. *CoRR*, abs/2112.01488, 2021.

[4] Thibault Formal, Carlos Lassance, Benjamin Piwowarski, and Stéphane Clinchant. SPLADE v2: Sparse lexical and expansion model for information retrieval. *CoRR*, abs/2109.10086, 2021.

[5] Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D. Manning. RAPTOR: recursive abstractive processing for tree-organized retrieval. *CoRR*, abs/2401.18059, 2024.

[6] Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables, 2017.

[7] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In *Advances in Neural Information Processing Systems*, volume 33, pages 5776–5788. Curran Associates, 2020.

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