A Comparative Study of Text Retrieval Models on DaReCzech

Jakub Stetina[1], Martin Fajcik[1], Michal Štefánik[2], and Michal Hradis[1]

[1] Faculty of Information Technology
Brno University of Technology, Czech Republic
[2] Faculty of Informatics
Masaryk University, Czech Republic
{xsteti05,ifajcik,ihradis}@fit.vutbr.cz, stefanik.m@mail.muni.cz

Abstract. This article presents a comprehensive evaluation of 7 off-the-shelf document retrieval models: Splade, Plaid, Plaid-X, SimCSE, Contriever, OpenAI ADA and Gemma2 chosen to determine their performance on the Czech retrieval dataset DaReCzech. The primary objective of our experiments is to estimate the quality of modern retrieval approaches in the Czech language. Our analyses include retrieval quality, speed, and memory footprint. Secondly, we analyze whether it is better to use the model directly in Czech text, or to use machine translation into English, followed by retrieval in English. Our experiments identify the most effective option for Czech information retrieval. The findings revealed notable performance differences among the models, with Gemma22 achieving the highest precision and recall, while Contriever performing poorly. Conclusively, SPLADE and PLAID models offered a balance of efficiency and performance.

Keywords: Information Retrieval, Evaluation, Comparison, Czech Language, Performance Assessment, Document Retrieval, Model Analysis.

1 Introduction

Information retrieval (IR) is used in areas such as search engines and question-answering systems. Lately, we've seen advancements in IR models [12,32,16, *inter alia*], but picking the right one for a non-English document collection can be challenging. We address this gap for Czech language by doing a comprehensive comparison in our study. In particular, we utilize DareCzech, a Czech retrieval and ranking dataset [14], for testing IR models to evaluate different IR models on Czech documents and queries. Our contributions are: (1) we analyze the index sizes to understand the storage requirements of various models, (2) we analyze the retrieval speed of such methods, to estimate how these models scale to large corpora in their default implementation, (3) we conduct ranking performance testing using multiple metrics on off-the-shelf models, and (4) we compare different model types, including those tested directly on the Czech

A. Horák, P. Rychlý, A. Rambousek (eds.): Proceedings of Recent Advances in Slavonic Natural Language Processing, RASLAN 2024, pp. 85–100, 2024. © Tribun EU 2024

dataset as well as on an English translation of the Czech dataset, keeping in mind the respective model's training data language, to provide insights into different approaches for indexing and retrieving czech. To the best of our knowledge, this is the first comparison study of existing state-of-the-art retrieval methods in the Czech language.

2 Related Work

Several well-known benchmarks have been used for evaluating information retrieval (IR) and text embedding models. **MS MARCO** [2] is widely used for passage and document retrieval, offering real-world web queries and answers. **MIRACL** [33] is a multilingual benchmark designed for retrieval across different languages. **MTEB** [18] provides a comprehensive evaluation across diverse tasks, including clustering, classification, and re-ranking.

Beyond English-language benchmarks, several datasets focus on IR evaluation within specific linguistic contexts. For Czech, the **CWRCzech** dataset [28] includes 100M query-document pairs based on Czech click data from *Seznam.cz* search logs. German-language IR is explored through **DPR German** [19] and **German LEGAL IR** [29], which assess retrieval in general and legal domains. The **SKQuad** [17] dataset provides an IR benchmark specifically for the Slovak language and the **Scandinavian Embedding Benchmark (SEB)** [7] provides a comprehensive evaluation framework for text embeddings in Scandinavian languages. Within MTEB, Polish [23] and Chinese [30] datasets extend the evaluation to language-specific IR tasks.

The dataset utilized in our study is **DaReCzech** (see Subsection 4.1), introduced in [14], which is specifically tailored for the Czech language and consists of manually annotated query-document pairs. The relevance annotations in DaReCzech are not binary, allowing for a more nuanced evaluation of relevance ranking models and enabling the use of various evaluation metrics.

3 Model Descriptions

BM25[24]. BM25 is a traditional lexical approach which has been widely used and had been the standard before the rise of neural models. It ranks documents based on a query's term frequency, inverse document frequency, and document length, meaning the importance of each term in the query and document is considered along with the document's length normalization, to produce relevance scores for each document.

In our study, we employed a BM25 baseline to assess the effectiveness of the other models. This model stands out as the only model with lemmatization applied to the query and document content, a distinction arising from the nature of BM25, which is not a neural model and relies on the precise lexical form of terms within the corpus.

For the Czech language, which features word inflection, lemmatization is essential for precise term matching and relevance ranking. Therefore, the lemmatized version of the corpus for BM25 is required.

splade-cocondenser-ensembledistil (SPLADE) [9]. (Sparse Lexical and Expansion Model) leverages sparse vocabulary-sized representations to leverage the advantages of BOW (bag-of-words). Splade operates by first applying a linear transformation to the BERT [6] output embeddings, then performing a dot product with the token embeddings from the whole vocabulary, resulting in a matrix of scores where each input token has a score for every token in the vocabulary[3]. While its predecessor, SparTerm [1], used a learned binary mask to select relevant scores from this matrix, Splade induces sparsity through a combination with a FLOPS regularizer and a logarithmic function during the representation computation. The final representation is obtained by summing the weights along the sequence tokens, producing a sparse embedding with a dimensionality equal to the vocabulary size. The second version of Splade improves this pooling mechanism to instead use the max for each token from the vocabulary. The model used in this comparison is the highest performing distilled version of splade as described in [8].

ColbertV2.0 (PLAID) [25]. The PLAID model represents a multi-vector approach. It extends the late interaction mechanism used in **ColBERT** [13] to enhance efficiency in information retrieval. In the original version of ColBERT, the comparison of query-document embeddings was performed by matching every token in the document embedding with every token in the query embedding, calculating scores using a maximum similarity function, where the highest similarity value for each query token across all document tokens is retained. ColBERTv2 [26] improved upon this approach by clustering the document embeddings into centroid clusters, thereby enabling a more efficient retrieval process. At search time, a fixed number of candidate clusters are selected, and their embeddings are decompressed to compute the final similarity scores.

In addition, **PLAID**, further enhances efficiency and performance by introducing a multi-stage candidate generation process. This approach includes steps for pruning and centroid-based interactions, progressively narrowing down the set of candidate passages. The final, smaller set of potential passages is then scored, resulting in a more streamlined and scalable retrieval pipeline. The model used in this study was trained on the English MS MARCO.

Plaidx-xlmr-large-mlir-neuclir (PLAID-X) [20,31]. Multilingual version of PLAID, PLAID-X builds upon the ColBERT architecture and employs a multilingual encoder XLM-RoBERTa (XLM-R) for multilingual and cross-lingual encodings. The model used in this study was trained using the translate-train approach on Chinese, Persian, and Russian data, relying on the XLM-R encoder for cross-language mappings.

[3] It utilizes the already pretrained masked language modeling head.

Text-embedding-ada-002 (OpenAI ADA) [22]. A closed-source model, that uses cosine similarity to compare two embeddings to calculate the resulting score.

Contriever-msmarco (Contriever) [10]. Contriever is a dense retrieval model that leverages self-supervised contrastive learning to effectively learn representations for information retrieval tasks. The model distinguishes between positive and negative passage pairs, where positive pairs are generated through independent window cropping from the original context (a document), and random token deletion. These approaches ensure that positive pairs share semantic content while exhibiting variation in phrasing, but also occasionally retaining lexical overlap. Negative pairs, on the other hand, are mined using MoCo[11], a method that builds a queue and uses a slowly changing encoder to generate negative samples. The model updates its document encoder by incorporating an online average of past parameters, ensuring that representations remain consistent across nearby training steps, hence making old representations stored in the queue compatible. This self-supervised training approach enables Contriever to produce dense vector representations for queries and documents, facilitating efficient retrieval and robust generalization across various retrieval tasks without the need for labeled datasets. The model used in this study was further finetuned on the English MS MARCO [2].

Simcse-dist-mpnet-paracrawl-cs-en (SimCSE)[10,4]. SimCSE employs contrastive learning to generate sentence embeddings, using simple dropout-based noise to create positive pairs from the same sentence while drawing negative pairs from other sentences within the batch. This approach trains the model to capture semantic similarities and differences between sentences without relying on large supervised datasets. Positive pairs are formed through augmentation techniques, such as random token deletion, replacement and masking, which introduce variability while preserving the original meaning. We chose to specifically test a model trained on the ParaCrawl [3] dataset (`SimCSE-Dist-MPNet-ParaCrawl`) as it achieved the highest DaReCzech performance at P@10 in the original work. The model used in this study was pretrained using an undisclosed Czech dataset from Seznam.cz and distilled on `czeng20-csmono` [15] and `Paracrawler v9` [3].

BGE Multilingual Gemma2 (Gemma2) [5]. BGE-Multilingual-Gemma2 is a large-language model based multilingual embedding model. It is directly finetuned using contrastive objective on an undisclosed diverse set of languages and tasks based on `google/gemma-2-9b` model [27]. During evaluation, the prompt used was: "Given a web search query, retrieve relevant passages that answer the query," as outlined in the instructions [4].

[4] `https://huggingface.co/BAAI/bge-multilingual-gemma2`

4 Experimental Setup

4.1 DaReCzech Dataset

DaReCzech is a Czech dataset designed for text relevance ranking, comprising over 1.6 million query-document pairs. It is divided into Train-big (1.4M pairs for model training), Train-small (97K pairs for model training), Dev (41K pairs), and Test (64K pairs), with no overlap between splits. Each record includes a query, URL, document title, document body text extract (BTE), and a relevance label. Queries are real user inputs, with minor corrections, and the documents are preprocessed to exclude irrelevant sections, ensuring a cleaner representation of content for ranking tasks.

We utilized DaReCzech by selecting test queries along with their associated relevant documents and additional documents to create a 100,000-document sample for indexing. This approach allowed us to maintain a representative document pool without indexing the entire dataset, primarily due to computational and economical overhead. Specifically, for the OpenAI Ada model, embedding generation incurs a cost. This balanced approach enabled a comprehensive evaluation while managing resource expenditure effectively. For relevance scores, we classified documents with scores above 0 as relevant and those with a score of 0 as non-relevant, aligning with binary metrics like precision and recall. More about the evaluation criteria can be found in Appendix A.1.

4.2 BM25 grid search

For fair comparison, we ran a grid search on the development set within our corpus to find the most optimal setting of the BM25's hyperparameters. The performance of the BM25 model was most optimal when the document length normalization parameter B was set to its maximum value of 1.0. This adjustment highlighted the importance of document length normalization in our particular case. The K_1 parameter, saturation of term frequency, showed minimal impact on our corpus, suggesting that its tuning had little to no effect on the performance. Based on this experiment, the BM25 hyperparameters were set to $[K_1, B] = [2, 1]$.

4.3 Dataset Translation

Some of the models tested were primarily or exclusively trained on English data. To achieve optimal performance and ensure a fair comparison, we applied document-level translation to the DaReCzech corpus, translating it into English using OPUS-MT, a multilingual translation model based on the OPUS corpora [5]. This approach allowed us to evaluate all models in their supported language setting.

[5] OPUS-MT translation model: `https://huggingface.co/Helsinki-NLP/opus-mt-cs-en`.

4.4 Segmentation

For the purpose of our evaluation, we employed a common indexing methodology across all models. For an initial experiment, we indexed documents in two ways: using only a truncated section up to each model's maximum input limit, and as multiple overlapping segments for longer documents, thus running the evaluation with two separate indices for each model[6]. However, since the overlapping approach did not yield any significant improvement, as can be seen in Figures 1 and 2, the later tested models were evaluated using non-overlapping segments only. The overlapping segments revealed an inherent bias of DaReCzech, as all the important data were usually concentrated at the beginning of the documents. This was also indicated by our extra analysis in Appendix C, making the cutoff method with no overlap sufficient. The cutoff lengths for each model were derived from the respective model papers, and a stride (if used) was selected to be roughly one-third of the maximum token length for each model as can be seen in Table 1.

Table 1: Overview of Experimental Model Configurations. All models were tested using truncated inputs, with select models additionally tested using segmented document inputs. The Segmenting column specifies the maximum token length and overlap window used for these experiments. OpenAI Ada is an only closed-access model we tested. (*The SPLADE output dimension represents the average number of tokens present in the output vector.).

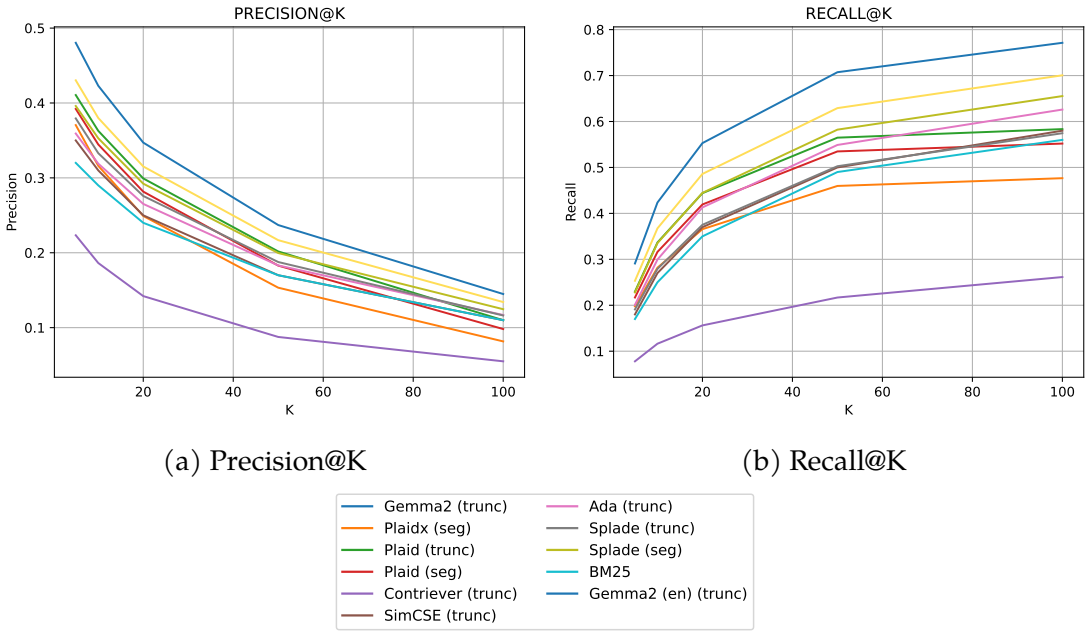
Model	Output Dim.	Max Tokens	Lang.	Segmenting
Splade	*45.7	256	en	256/86
PLAID	128	300	en	300/100
PLAID-X	128	180	cs	180/60
Contriever	768	256	en	–
SimCSE	256	128	cs	–
Gemma2	3584	512	cs/en	–
OpenAI ada	1536	8192	cs	–

5 Results and Analysis

The precision and recall values (Figures 1a,1b) for all models across different k values exhibit distinct patterns, with Contriever performing the worst, even below BM25. The stricly top-performing model is Gemma2. Notably, both the Czech and English versions of Gemma2 rank highly, with the Czech model

[6] As a result, the evaluation with overlapping made the effective number of document represenations in the (single-vector) indices exceed the base count of 100,000 documents. During the retrieval process, all the duplicate versions of documents were removed leaving only the highest rank of the same document.

showing a slight advantage in performance. Beneath Gemma2, the best results come from the PLAID models. However, segmenting the documents with these models demonstrates a decline in both precision and recall as k increases, possibly due to an accumulation of irrelevant information from segment-level retrievals impacting the overall ranking quality[7].

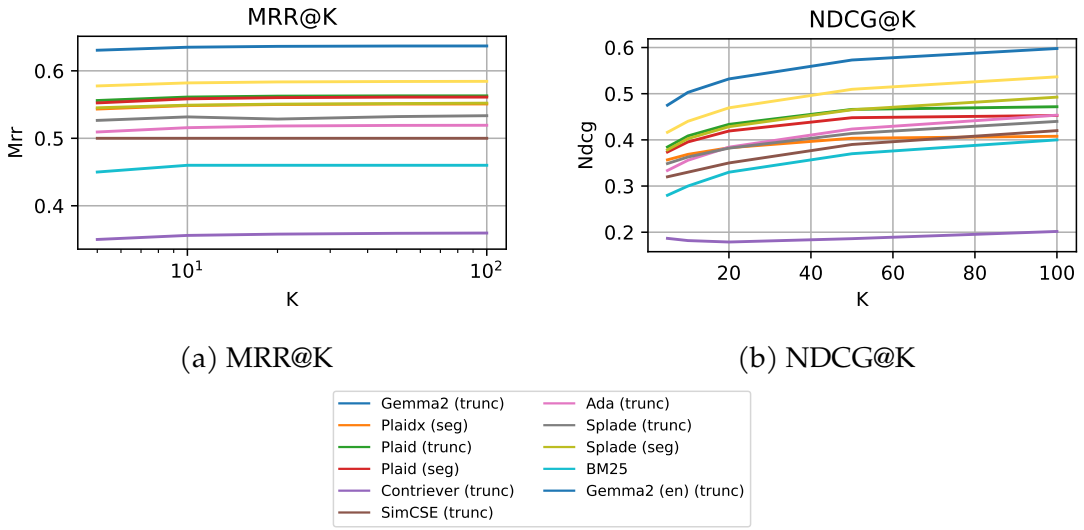


(a) Precision@K (b) Recall@K

Fig. 1: Comparison of Precision and Recall at different values of k.

These observed trends are even more evident in the full MRR and NDCG metrics (Figures 2a 2b), where the differences among models are more pronounced. In the MRR graphs, nearly consistent performance across different k values indicates that while the general retrieval ability remains stable, the ranking quality of results varies significantly between models. The superior performance of Gemma2 and the relative weaknesses of Contriever are reflected here, reinforcing the patterns observed in the precision and recall figures. This alignment suggests that models with higher precision and recall also exhibit better ordering and ranking capabilities, as demonstrated by their MRR and NDCG scores.

[7] The evaluation was conducted in two ways for some models. In one the retrieval function retrieve(k) was called for each tested value of k separately and in the other the highest tested value of k was chosen and then the results cut off down to the tested k value. This was done to especially examine the PLAID retrieval implementation, which determines different hyperparameters for its approximate nearest-neighbor search for different k value. This however did not show any significant change in the tested metrics (in these cases the better results were kept).



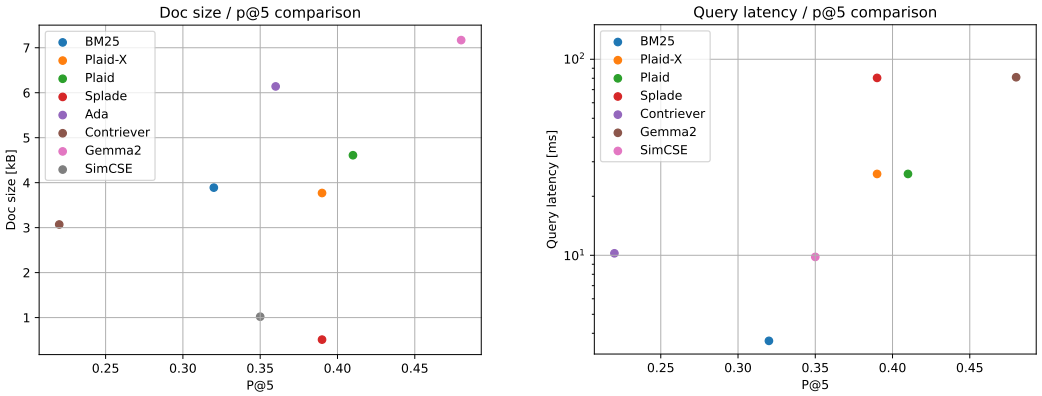
(a) MRR@K (b) NDCG@K

Fig. 2: Comparison of MRR, NDCG at different values of k.

Figure 3a demonstrates the trade-off between document size (estimated by averaging the size of each index by the number of indexed documents) and retrieval precision. As anticipated, BM25 maintains a compact document representation but exhibits a low Precision@5 performance, with Contriever faring even worse. PLAID-X achieves modest gains over BM25, with a smaller index size per document due to a restrictive 180-token limit. SPLADE, while comparable to PLAID-X in precision, maintains a much smaller index thanks to its sparse nature [8]. The original PLAID model, without cross-lingual settings, slightly outperforms both PLAID-X and SPLADE, though it incurs a larger index size due to a higher token limit of 300. OpenAI's Ada model struggles to compete, hindered by its large embedding dimension, resulting in a substantial index size that does not justify its middling performance. The Gemma2 model emerges as the top performer, albeit with the largest embedding size, indicating a trade-off between high retrieval accuracy and storage requirements. Such result is aligned with observations in [21], where authors demonstrate that embedding performance tends to scale with model size and embedding dimension.

An analysis of query latency in Figure 3b shows that BM25 achieves the fastest query times, which aligns with its straightforward term-matching approach. Contriever and SimCSE, both using single-vector embeddings and cosine similarity, follow closely. The PLAID-X and PLAID models exhibit slightly longer latencies, likely due to their multi-stage retrieval process, which involves candidate selection and more complex ranking steps, contributing to a moderate increase in query time. SPLADE and Gemma2 are slower still; SPLADE's

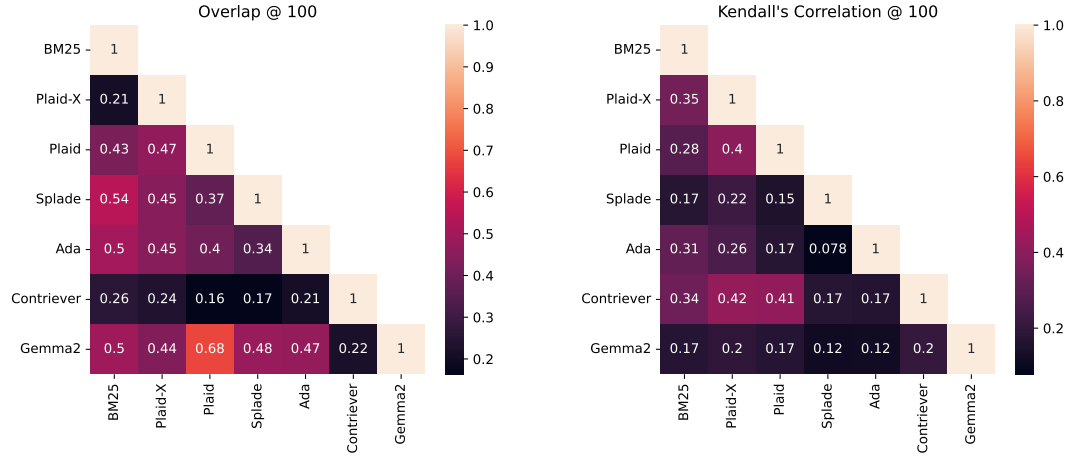
[8] Some models achieve similar or even significantly smaller index sizes compared to BM25 due to truncation; BM25 indexed entire documents without truncation, while many other models were limited to a few hundred tokens per document. This limit, especially for longer documents, led to reduced overall index sizes.



(a) Doc repr. size / P@5 comparison (b) Query latency / P@5 comparison

Fig. 3: Doc size, query latency in relation to P@5.

sparse representation requires additional computation to dynamically calculate sparse scores, while Gemma2's high-dimensional embeddings impose added processing overhead. These patterns suggest that models with multi-stage or complex scoring mechanisms naturally incur higher latency compared to more direct embedding or term-based approaches.



(a) Overlap @ 100 (b) Kendall's Correlation @ 100

Fig. 4: Pairwise overlap and correlation of overlapped items in top-100 responses of different IR systems.

Regarding the overlap among the models, as depicted in Figure 4, Contriever consistently exhibits the lowest overlap scores across comparisons with other models, a finding that aligns well with its previously observed underperfor-

mance in Figures 1a and 1b. Notably, PLAID and PLAID-X display a high degree of overlap and strong Kendall τ correlation, likely attributable to their shared architecture and training approach, with PLAID-X being a multilingual adaptation of PLAID. Interestingly, we also observe a notably high overlap value between PLAID and GEMMA, which could be attributed to GEMMA's training on diverse multilingual data that likely includes features common to PLAID's retrieval methodology.

6 Conclusion

In this paper, we evaluated 7 off-the-shelf information retrieval models on the DaReCzech corpus, comparing their performance against the traditional BM25 approach. The goal was to identify the most effective model for information retrieval in Czech.

Our findings showed that Gemma2 consistently delivered the best precision and recall metrics across various k values, with the Czech version slightly outperforming the English one. However, its high retrieval accuracy came with a large index size due to high-dimensional embeddings exceeding even the multi-vector models. In contrast, BM25 and Contriever exhibited the poorest performance, with Contriever notably underperforming and struggling to match BM25's baseline.

SPLADE and the PLAID models offered a balance between performance and efficiency. SPLADE's sparse representation resulted in the smallest index size, making it suitable for resource-constrained applications, while the PLAID models, especially the original, provided higher precision with modest increases in index size. The ColBERT-based models performed well unsegmented, but segmenting for long documents led to a decrease in performance as k increased.

For Czech-language IR tasks, Gemma2 is recommended if accuracy is the top priority and storage is manageable. SPLADE is a practical choice when memory efficiency is crucial, and PLAID/PLAID-X offer a middle ground, particularly with token limit adjustments. This study underscores the trade-offs between model complexity, storage, and retrieval quality, guiding suitable model selection for Czech-language IR.

Acknowledgements. This work was supported by project Ministry of Culture of the Czech Republic through NAKI III project semANT, grant. no DH23P03OVV060, Horizon EU programme through project ELOQUENCE, grant no. 101135916, and by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90254).

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Bai, Y., Li, X., Wang, G., Zhang, C., Shang, L., Xu, J., Wang, Z., Wang, F., Liu, Q.: Sparterm: Learning term-based sparse representation for fast text retrieval (2020), https://arxiv.org/abs/2010.00768
2. Bajaj, P., Campos, D., Craswell, N., Deng, L., Gao, J., Liu, X., Majumder, R., McNamara, A., Mitra, B., Nguyen, T., Rosenberg, M., Song, X., Stoica, A., Tiwary, S., Wang, T.: Ms marco: A human generated machine reading comprehension dataset (2018), https://arxiv.org/abs/1611.09268
3. Bañón, M., Chen, P., Haddow, B., Heafield, K., Hoang, H., Esplà-Gomis, M., Forcada, M.L., Kamran, A., Kirefu, F., Koehn, P., Ortiz Rojas, S., Pla Sempere, L., Ramírez-Sánchez, G., Sarrías, E., Strelec, M., Thompson, B., Waites, W., Wiggins, D., Zaragoza, J.: ParaCrawl: Web-scale acquisition of parallel corpora. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J. (eds.) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 4555–4567. Association for Computational Linguistics, Online (Jul 2020). https://doi.org/10.18653/v1/2020.acl-main.417, https://aclanthology.org/2020.acl-main.417
4. Bednář, J., Náplava, J., Barančíková, P., Lisický, O.: Some like it small: Czech semantic embedding models for industry applications (2023), https://arxiv.org/abs/2311.13921
5. Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., Liu, Z.: Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation (2024), https://arxiv.org/abs/2402.03216
6. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2019), https://arxiv.org/abs/1810.04805
7. Enevoldsen, K., Kardos, M., Muennighoff, N., Nielbo, K.L.: The scandinavian embedding benchmarks: Comprehensive assessment of multilingual and monolingual text embedding (2024), https://arxiv.org/abs/2406.02396
8. Formal, T., Lassance, C., Piwowarski, B., Clinchant, S.: From distillation to hard negative sampling: Making sparse neural ir models more effective (2022), https://arxiv.org/abs/2205.04733
9. Formal, T., Piwowarski, B., Clinchant, S.: Splade: Sparse lexical and expansion model for first stage ranking (2021), https://arxiv.org/abs/2107.05720
10. Gao, T., Yao, X., Chen, D.: Simcse: Simple contrastive learning of sentence embeddings (2022), https://arxiv.org/abs/2104.08821
11. He, K., Fan, H., Wu, Y., Xie, S., Girshick, R.: Momentum contrast for unsupervised visual representation learning (2020), https://arxiv.org/abs/1911.05722
12. Hendrycks, D., Burns, C., Basart, S., Zou, A., Mazeika, M., Song, D., Steinhardt, J.: Measuring massive multitask language understanding (2021), https://arxiv.org/abs/2009.03300
13. Khattab, O., Zaharia, M.: Colbert: Efficient and effective passage search via contextualized late interaction over BERT. CoRR **abs/2004.12832** (2020), https://arxiv.org/abs/2004.12832
14. Kocián, M., Náplava, J., Štancl, D., Kadlec, V.: Siamese bert-based model for web search relevance ranking evaluated on a new czech dataset (2021), https://arxiv.org/abs/2112.01810
15. Kocmi, T., Popel, M., Bojar, O.: Announcing czeng 2.0 parallel corpus with over 2 gigawords. arXiv preprint arXiv:2007.03006 (2020)

16. Koto, F., Li, H., Shatnawi, S., Doughman, J., Sadallah, A.B., Alraeesi, A., Almubarak, K., Alyafeai, Z., Sengupta, N., Shehata, S., Habash, N., Nakov, P., Baldwin, T.: Arabicmmlu: Assessing massive multitask language understanding in arabic (2024), https://arxiv.org/abs/2402.12840
17. Technical University of Košice, D.o.E., Communication, M.: Retrieval-skquad dataset (2023), https://huggingface.co/datasets/TUKE-KEMT/retrieval-skquad, dataset for Slovak search retrieval evaluation, licensed under CC BY-NC-SA 4.0
18. Muennighoff, N., Tazi, N., Magne, L., Reimers, N.: Mteb: Massive text embedding benchmark (2023), https://arxiv.org/abs/2210.07316
19. Möller, T., Risch, J., Pietsch, M.: Germanquad and germandpr: Improving non-english question answering and passage retrieval (2021), https://arxiv.org/abs/2104.12741
20. Nair, S., Yang, E., Lawrie, D., Duh, K., McNamee, P., Murray, K., Mayfield, J., Oard, D.W.: Transfer learning approaches for building cross-language dense retrieval models. In: Proceedings of the 44th European Conference on Information Retrieval (ECIR) (2022), https://arxiv.org/abs/2201.08471
21. Neelakantan, A., Xu, T., Puri, R., Radford, A., Han, J.M., Tworek, J., Yuan, Q., Tezak, N., Kim, J.W., Hallacy, C., et al.: Text and code embeddings by contrastive pre-training. arXiv preprint arXiv:2201.10005 (2022)
22. OpenAI: Openai ada model for retrieval. https://platform.openai.com/docs/models/embeddings (2023), accessed: 2024-10-29
23. Poświata, R., Dadas, S., Perełkiewicz, M.: Pl-mteb: Polish massive text embedding benchmark (2024), https://arxiv.org/abs/2405.10138
24. Robertson, S., Zaragoza, H.: The probabilistic relevance framework: Bm25 and beyond. Foundations and Trends in Information Retrieval **3**, 333–389 (01 2009). https://doi.org/10.1561/1500000019
25. Santhanam, K., Khattab, O., Potts, C., Zaharia, M.: Plaid: An efficient engine for late interaction retrieval (2022), https://arxiv.org/abs/2205.09707
26. Santhanam, K., Khattab, O., Saad-Falcon, J., Potts, C., Zaharia, M.: Colbertv2: Effective and efficient retrieval via lightweight late interaction (2022), https://arxiv.org/abs/2112.01488
27. Team, G., Riviere, M., Pathak, S., Sessa, P.G., Hardin, C., Bhupatiraju, S., Hussenot, L., Mesnard, T., Shahriari, B., Ramé, A., Ferret, J., Liu, P., Tafti, P., Friesen, A., Casbon, M., Ramos, S., Kumar, R., Lan, C.L., Jerome, S., Tsitsulin, A., Vieillard, N., Stanczyk, P., Girgin, S., Momchev, N., Hoffman, M., Thakoor, S., Grill, J.B., Neyshabur, B., Bachem, O., Walton, A., Severyn, A., Parrish, A., Ahmad, A., Hutchison, A., Abdagic, A., Carl, A., Shen, A., Brock, A., Coenen, A., Laforge, A., Paterson, A., Bastian, B., Piot, B., Wu, B., Royal, B., Chen, C., Kumar, C., Perry, C., Welty, C., Choquette-Choo, C.A., Sinopalnikov, D., Weinberger, D., Vijaykumar, D., Rogozińska, D., Herbison, D., Bandy, E., Wang, E., Noland, E., Moreira, E., Senter, E., Eltyshev, E., Visin, F., Rasskin, G., Wei, G., Cameron, G., Martins, G., Hashemi, H., Klimczak-Plucińska, H., Batra, H., Dhand, H., Nardini, I., Mein, J., Zhou, J., Svensson, J., Stanway, J., Chan, J., Zhou, J.P., Carrasqueira, J., Iljazi, J., Becker, J., Fernandez, J., van Amersfoort, J., Gordon, J., Lipschultz, J., Newlan, J., yeong Ji, J., Mohamed, K., Badola, K., Black, K., Millican, K., McDonell, K., Nguyen, K., Sodhia, K., Greene, K., Sjoesund, L.L., Usui, L., Sifre, L., Heuermann, L., Lago, L., McNealus, L., Soares, L.B., Kilpatrick, L., Dixon, L., Martins, L., Reid, M., Singh, M., Iverson, M., Görner, M., Velloso, M., Wirth, M., Davidow, M., Miller, M., Rahtz, M., Watson, M., Risdal, M., Kazemi, M., Moynihan, M., Zhang, M., Kahng, M., Park, M., Rahman, M., Khatwani, M., Dao, N., Bardoliwalla, N., Devanathan, N., Dumai, N., Chauhan, N., Wahltinez, O., Botarda,

P., Barnes, P., Barham, P., Michel, P., Jin, P., Georgiev, P., Culliton, P., Kuppala, P., Comanescu, R., Merhej, R., Jana, R., Rokni, R.A., Agarwal, R., Mullins, R., Saadat, S., Carthy, S.M., Cogan, S., Perrin, S., Arnold, S.M.R., Krause, S., Dai, S., Garg, S., Sheth, S., Ronstrom, S., Chan, S., Jordan, T., Yu, T., Eccles, T., Hennigan, T., Kocisky, T., Doshi, T., Jain, V., Yadav, V., Meshram, V., Dharmadhikari, V., Barkley, W., Wei, W., Ye, W., Han, W., Kwon, W., Xu, X., Shen, Z., Gong, Z., Wei, Z., Cotruta, V., Kirk, P., Rao, A., Giang, M., Peran, L., Warkentin, T., Collins, E., Barral, J., Ghahramani, Z., Hadsell, R., Sculley, D., Banks, J., Dragan, A., Petrov, S., Vinyals, O., Dean, J., Hassabis, D., Kavukcuoglu, K., Farabet, C., Buchatskaya, E., Borgeaud, S., Fiedel, N., Joulin, A., Kenealy, K., Dadashi, R., Andreev, A.: Gemma 2: Improving open language models at a practical size (2024), https://arxiv.org/abs/2408.00118

28. Vonásek, J., Straka, M., Krč, R., Lasonová, L., Egorova, E., Straková, J., Náplava, J.: Cwrczech: 100m query-document czech click dataset and its application to web relevance ranking. In: Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval. SIGIR 2024, vol. 38, p. 1221–1231. ACM (Jul 2024). https://doi.org/10.1145/3626772.3657851, http://dx.doi.org/10.1145/3626772.3657851
29. Wrzalik, M., Krechel, D.: GerDaLIR: A German dataset for legal information retrieval. In: Aletras, N., Androutsopoulos, I., Barrett, L., Goanta, C., Preotiuc-Pietro, D. (eds.) Proceedings of the Natural Legal Language Processing Workshop 2021. pp. 123–128. Association for Computational Linguistics, Punta Cana, Dominican Republic (Nov 2021). https://doi.org/10.18653/v1/2021.nllp-1.13, https://aclanthology.org/2021.nllp-1.13
30. Xiao, S., Liu, Z., Zhang, P., Muennighoff, N., Lian, D., Nie, J.Y.: C-pack: Packed resources for general chinese embeddings (2024), https://arxiv.org/abs/2309.07597
31. Yang, E., Lawrie, D., Mayfield, J., Oard, D.W., Miller, S.: Translate-distill: Learning cross-language dense retrieval by translation and distillation. In: Proceedings of the 46th European Conference on Information Retrieval (ECIR) (2024), https://arxiv.org/abs/2401.04810
32. Yüksel, A., Köksal, A., Şenel, L.K., Korhonen, A., Schütze, H.: Turkishmmlu: Measuring massive multitask language understanding in turkish (2024), https://arxiv.org/abs/2407.12402
33. Zhang, X., Thakur, N., Ogundepo, O., Kamalloo, E., Alfonso-Hermelo, D., Li, X., Liu, Q., Rezagholizadeh, M., Lin, J.: Making a miracl: Multilingual information retrieval across a continuum of languages (2022), https://arxiv.org/abs/2210.09984

A Evaluation Process

A.1 Evaluation Metrics

To assess the performance of these IR models, we employ a range of standard evaluation metrics:

Precision Precision quantifies the accuracy of relevant documents in the retrieved set and is defined as:

$$\text{Precision} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{retrieved documents}\}|} \tag{1}$$

Recall Recall measures the ability of the model to retrieve all relevant documents from the corpus and is given by:

$$\text{Recall} = \frac{|\{\text{relevant documents}\} \cap \{\text{retrieved documents}\}|}{|\{\text{relevant documents in corpus}\}|} \tag{2}$$

MRR (Mean Reciprocal Rank) Mean Reciprocal Rank (MRR) evaluates the ranking quality by taking the mean of the reciprocal ranks of the first relevant document for each query:

$$\text{MRR} = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{\text{rank}_i} \tag{3}$$

where rank_i is the position of the first relevant document for the i-th query and $|Q|$ is the total number of queries.

MAP (Mean Average Precision) Mean Average Precision (MAP) calculates the average precision for each query and averages these scores across all queries, thereby reflecting the model's ranking consistency. For a given query q, the average precision is:

$$\text{AP}_q = \frac{1}{|\{\text{relevant documents for } q\}|} \sum_{k=1}^{N} \text{Precision}(k) \cdot \text{rel}(k) \tag{4}$$

where N is the total number of documents, $\text{Precision}(k)$ is the precision at rank k, and $\text{rel}(k)$ is a binary indicator of relevance at rank k. MAP is then:

$$\text{MAP} = \frac{1}{|Q|} \sum_{q=1}^{|Q|} \text{AP}_q. \tag{5}$$

nDCG (Normalized Discounted Cumulative Gain) Normalized Discounted Cumulative Gain (nDCG) evaluates the ranked list's quality by considering the position of relevant documents in the ranking. For a query q, nDCG at rank p is calculated as:

$$\text{DCG}_p = \sum_{k=1}^{p} \frac{2^{\text{rel}(k)} - 1}{\log_2(k+1)} \tag{6}$$

$$\text{nDCG}_p = \frac{\text{DCG}_p}{\text{IDCG}_p} \tag{7}$$

where $\text{rel}(k)$ is the relevance score of the document at rank k, and IDCG_p is the ideal DCG, obtained by sorting documents in the perfect order of relevance.

A.2 Additional Evaluation Criteria

In addition to the standard metrics, we also consider two specific criteria:

Representation Size Examining the memory footprint required by each model's document representations (measured per document as kB/doc),

Query Latency Query latency refers to the duration taken by an information retrieval system to retrieve and present relevant documents in response to a given query.

Kendall's τ Rank Correlation and Lexical Overlap assessing the consistency of ranking across the models on the top 100 retrieved results for each query - helps understanding how well the models agree on the most relevant documents

B BM25 Hyperparameter Tuning

We perform a BM25 grid search to tune the K1 and B parameters for optimal results on the corpus. The results from the grid search are visualized in Figure 5.

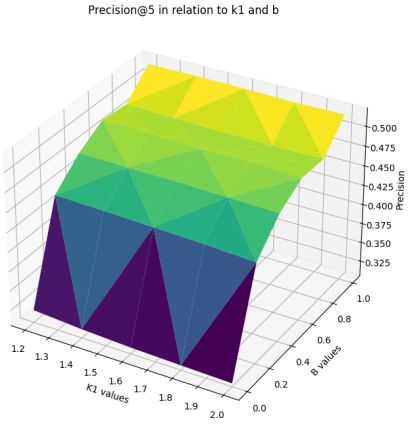


Fig. 5: BM25 hyperparameters grid search.

C ColBERT's Token-level Focus

To estimate which parts of the document are important, the study analyzed ColBERTv2, a model that uses a multi-vector approach, where each token in a document is represented by a separate vector. By examining the vectors of the retrieved documents, tokens with the most interaction with the query were identified. This might indicate which specific parts of the documents were most

relevant to the query and contributed to the retrieval process for the given document.

This experiment examined two modes for the document-query scores (samples with scores aggregated from Colbert's similarity matrix through max-pooling over query token representations (in contrast with the original Colbert's MaxSim operation which computes max-pooling over document token representations) can be seen below in Figure 6) visualized as a probability distribution using the softmax function with the brighter color denoting higher similarity score):

MaxSim operation: particularly chosen as it reflects how the model selects positive document-query pairs, and identifies the best score for each query token with the highest scoring document token, highlighting the most significant interactions.

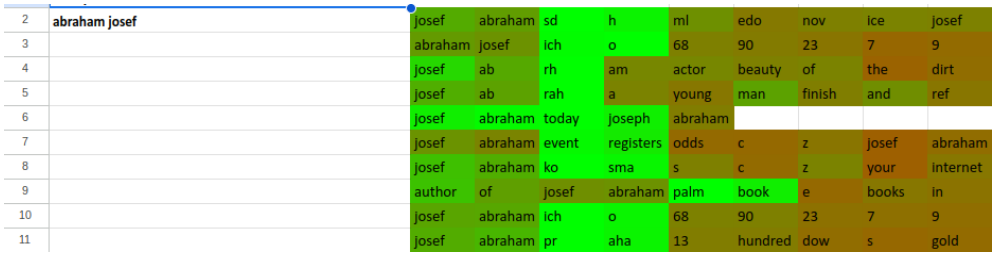


Fig. 6: Colbert interpretability.