Team OpenWebSearch at CLEF 2024: LongEval

Daria Alexander[1], Maik Fröbe[2], Gijs Hendriksen[1], Ferdinand Schlatt[2], Matthias Hagen[2], Djoerd Hiemstra[1], Martin Potthast[3] and Arjen P. de Vries[1]

[1]Radboud Universiteit Nijmegen
[2]Friedrich-Schiller-Universität Jena
[3]University of Kassel, hessian.AI, ScaDS.AI

Abstract

We describe the OpenWebSearch group's participation in the CLEF 2024 LongEval IR track. Our submitted runs explore how historical data from the past can be transferred into future retrieval systems. Therefore, we incorporate relevance information from past click logs into the query reformulation process via keyqueries and into the indexing process via a reverted index and ultimately incorporate both into learning-to-rank pipelines to ensure that retrieval is also possible for novel queries that were not seen before. Our evaluation shows that keyqueries substantially outperform other approaches for queries with historical click data available.

Keywords

learning-to-rank, query logs, keyqueries

1. Introduction

Historical data obtained from query logs may substantially help to improve the rankings of future retrieval models. The scenario of the LongEval retrieval task [1, 2, 3, 4, 5, 6, 7] aims to study this area where retrieval models have access to relevance labels estimated from past query logs with click models to provide effective rankings in the future. Especially queries that have been seen before, i.e., for which past relevance information is available, have a high potential to leverage past relevance information for highly effective rankings if the intent of the query did not drift. For example, under the most simple assumption that queries have the same intent and that documents did not change, almost perfect rankings can be derived by simply ordering documents for a query by their estimated relevance from past query logs. However, as query intents and document content might change substantially over time, this transfer of old relevance information to future retrieval tasks might not be straightforward.

We implement this relevance transfer for queries that overlap from past query logs to future retrieval tasks via two orthogonal concepts: (1) query reformulation with keyqueries, and (2) document reformulation. For the query reformulation, we leverage the concept of keyqueries [8, 9] that try, for a set of target documents, to identify the query that ranks the target documents highly while ensuring that the resulting query does not overfit on the target documents. For the document reformulation, we combined the concept of the corpus graph [10] with the concept of the reverted index [11]. Specifically, we identify which documents are highly similar to documents that were relevant to some query in the past (i.e., some form of a corpus graph construction) to subsequently index those documents with the queries to which they were relevant in the past (i.e., some form of a reverted index). If documents would not change their meaning and if queries would not change their intent, both concepts, the query reformulation and the document reformulation, would yield ideal rankings. Still, a realistic search engine would also need to produce good rankings for new queries or queries, respectively, documents that changed their content or meaning.

To address this problem and to generalize to new queries and potentially changed query intents, we incorporate our query and document reformulations into learning-to-rank models. Learning-to-Rank aims to identify a combination of features that produce an effective ranking [12]. Even in the era of pre-trained transformers [13], feature-based learning-to-rank remains important as it can integrate features not available in transformers, compensating for knowledge to which transformers have no

CLEF 2024: Conference and Labs of the Evaluation Forum, September 09–12, 2024, Grenoble, France

access [14, 15]. Especially commercial search engines might combine many features, e.g., a recent leak claims that Google search incorporates more than 14 000 features into their ranking.[1] Overall, we create a set of over 100 features derived from submissions to the Workshop on Open Web Search [16, 17] and combine them with learning-to-rank in our submissions. Our code and trained LambdaMART models are available online.[2]

2. Related Work

We review related work on redundancy in information retrieval setups, keyqueries, and the corpus graph and reverted index.

Redundancy in Information Retrieval Setups Normally, it is good practice to avoid redundancy between training, validation, and test splits in experiments, as otherwise, the effectiveness could be overestimated due to train–test leakage [18, 19]. Especially for IR experiments, redundant documents might cause effectiveness scores to be overestimated because retrieval models get a reward for showing the same model multiple times [20, 21]. Similar problems can occur for learned models that might overfit to redundancy in the training data [22]. However, in the LongEval scenario, redundancy emerges naturally, as queries and documents might overlap over time, which is no form of train–test leakage as the datasets are partitioned over time [1, 3]. In this setting, redundant data might be especially helpful, e.g., as previously showcased when relevance judgments were transferred from the ClueWeb09 corpus to ClueWeb12 via near-duplicate detection [23]. We follow this approach and transfer the relevance judgments to the newer dataset splits in the LongEval scenario via keyqueries and the corpus graph.

Keyqueries The concept of keyqueries [9] aims to formulate a query that retrieves a set of target documents at the top-positions and has been applied to scholarly search [24], medical search [25], privacy scenarios [26], etc. For a set D of documents, a query q is a *keyquery* against some retrieval system S, iff q fulfills the following three conditions [9]: (1) every $d \in D$ is in the top-k results returned by S for q, (2) q has at least l results, and (3) no $q' \subset q$ fulfills the first two conditions. The first two conditions (i.e., the parameters k and l) determine the desired specificity and the generality of a keyquery, while the third condition is a minimality constraint to avoid adding further terms to a query that already retrieves the target records at high ranks. Previous work applied this concept only to static corpora, but we now extend it to evolving corpora in the LongEval scenario.

Corpus Graph and Reverted Indexes The corpus graph [10] consists of nodes that correspond to documents in the corpus and edges that are formed based on the similarity of documents. This similarity is either lexical or semantic, and is used in a re-ranking scenario to also consider documents highly similar to the top-ranked documents to improve the recall [10]. The reverted index [11] directly stores which documents should be ranked for which queries. We combine both concepts in the LongEval scenario: by building a corpus graph between the documents that were relevant to some queries in the past to the documents in the current corpus, we index those documents into the reverted index.

3. Methodology

Our last year participation at LongEval was aimed at finding out whether generating multiple query variants for the same information need improves retrieval effectiveness. We generated query variants using ChatGPT and fused ranking results obtained with the original query and different query variants. We found out that query variant generation improves over time and follows the same trend as BM25

[1] https://sparktoro.com/blog/an-anonymous-source-shared-thousands-of-leaked-google-search-api-documents-with-me-everyone-in-seo-should-see-them

[2] https://github.com/OpenWebSearch/LONGEVAL-24

baseline, therefore the query variant generation showed its robustness [27]. Still, the improvements were only minor.

However, last year we did not explore the information that is provided by the documents in the past and whether this information can be useful for the future. Therefore, we decided to extend the queries with the terms for the relevant documents. Also, for the non-overlapping queries we wanted to have a system that does not rely on information from the past. For that we used a learning-to-rank approach, which utilizes features from the components submitted to the Workshop on Open Web Search (WOWS) [16].

3.1. Keyqueries

We noticed that queries overlap over different time slots, and in case their intent stays the same, we aim to transfer their relevance information to the new time slots. Consecutively, for those queries we know what documents were clicked a few months ago. We decided to use this feedback and query expansion with the BO1 model [28] to create keyqueries and use the same approach as [25]. Thereby, we use BO1 to obtain candidate terms for query terms, as pilot experiments showed that BO1 expansion terms yield higher effectiveness than RM3 [29] expansions. We inserted the clicked documents into the current corpus and reformulated the queries with the BO1 model until those documents were in the top positions. After that we removed old documents from the ranking. This implementation of the keyquery concept is not the most effective one, more effective approaches that leverage a generate-and-test paradigm [26] exist and are interesting directions for future work (i.e., explicitly generating many variants and selecting the variants that are highly effective).

3.2. Reverted index

First, we identify documents in the new corpus that are higly similar to documents that were relevant to documents that were relevant to some queries in the past data (we use all available past data). We find those candidate terms by building an index with PyTerrier for the new corpus and submitting every relevant document from the past to the new corpus to retrieve the 10 nearest neighbors according to BM25. We then create a reverted index by indexing the document of position 1 with 10 times the query to which a document was relevant, the document on position 2 with 9 times, etc. For the final retrieval, we use BM25 against this constructed reverted index.

3.3. Learning to Rank

While a large share of the queries in the test collections have overlap with the queries in the training splits, this is of course not the case for all queries. Hence, we also needed a system that could be used when information from the past could not be exploited directly. For these cases, we also developed a simple learning-to-rank approach, which used features from a large number of components submitted to the Workshop on Open Web Search (WOWS) [16].

For our learning-to-rank systems, we re-ranked the top 100 BM25 results using LambdaMART [30]. We implemented our pipelines with PyTerrier [31], using LightGBM [32] for the LambdaMART implementation. The feature extraction components were all executed in TIREx [33] once, after which their outputs were cached for easy repeated experimentation.

We split the 2024 training set into a training and validation split, which we used to tune LightGBM's hyperparameters. We performed several runs, each with different subsets of features:

ows-ltr-wows-base-rerank Query-only scores (QPP scores [34], classified intents [35], and health-relatedness [36]); document-only scores (health-relatedness [36], classified genre [37], and readability scores [37]); and lexical matching models built into PyTerrier (BM25, PL2, DirichletLM, DLH, and LGD).

Table 1
The effectiveness of the seven submitted runs and the BM25 baseline on the June and August 2023 test sets, respectively. We report the nDCG and the nDCG@10 as well as nDCG and nDCG@10 when unjudged documents are removed (Cond. nDCG and Cond. nDCG@10)

Approach / Run	nDCG		nDCG@10		Cond. nDCG		Cond. nDCG@10	
	June	August	June	August	June	August	June	August
ows-bm25-bo1-keyqueries	**0.332**	**0.242**	0.240	0.190	**0.471**	**0.350**	**0.448**	**0.343**
ows-bm25-reverted-index	0.305	0.228	**0.241**	**0.192**	0.400	0.307	0.390	0.305
ows-ltr-all	0.293	0.223	0.226	0.186	0.395	0.305	0.384	0.303
ows-ltr-wows-rerank-and-reverted-index	0.289	0.212	0.219	0.172	0.402	0.305	0.393	0.302
ows-ltr-wows-rerank-and-keyquery	0.283	0.216	0.212	0.176	0.396	0.306	0.386	0.303
ows-ltr-wows-all-rerank	0.245	0.204	0.155	0.158	0.389	0.304	0.378	0.301
ows-ltr-wows-base-rerank	0.239	0.177	0.151	0.120	0.390	0.301	0.378	0.298
BM25	0.252	0.191	0.166	0.141	0.388	0.300	0.375	0.297

ows-ltr-wows-all-rerank The features from **ows-ltr-wows-base-rerank** plus additional, neural-based query-document scores (RankZephyr [38], Sparse Cross Encoder [39], LiT5 [40], SBERT [41], MonoT5 [42], ColBERT [43], and ANCE [44]).

ows-ltr-wows-rerank-and-reverted-index The features from **ows-ltr-wows-all-rerank**, plus three features related to the reverted index: 1) whether the query-document pair has been encountered in the past, 2) the maximum score for this query-document pair in the past, and 3) the mean score for this query-document pair in the past.

ows-ltr-wows-rerank-and-keyquery The features from **ows-ltr-wows-all-rerank**, plus two keyquery-related features: 1) whether this query-document pair has been encountered in the keyquery run, and 2) the score of this query-document pair in the keyquery run.

ows-ltr-all A combination of all features described above.

Note that some of the features – especially the neural query-document features – can be prohibitively expensive to compute in a real-world system. Our learning-to-rank results thus indicate the theoretical performance of a system using all of these models together, while in practice, a system might only use a small subset of them.

4. Results

We will evaluate our submitted runs on all queries and on only overlapping queries.

4.1. Results for all queries and overlapping queries

We report the nDCG [45] without cutoff and at a cutoff at 10, and condensed variants where all unjudged documents are removed [46] (although this better handles the effects of unjudged documents than dedicated measures like Bpref [46], it is known to overestimate the effectiveness [47] which was only recently confirmed [48]). The share of unjudged documents is 68-77% for June and 73-82% for August 2023 (cutoff at 10) depending on the runs.

Table 1 shows that most of the runs outperform the baseline, with the baseline never being the best approach. It is a big improvement in comparison to the last year when the baseline was still the best approach for several runs. We can see that using keyqueries outperforms other approaches along with the reverted index for nDCG@10.

In Table 2 we present the results for the queries that are overlapping between January, June and August. Overall, our scores are higher when considering only overlapping queries rather than all queries. We can observe that utilising the information from the past click logs is beneficial especially for

Table 2
The effectiveness of the seven submitted runs and the BM25 baseline on queries that overlap between January 2023 train set, June 2023 test set and August 2023 test set: 126 queries in June and 141 queries in August. We report the nDCG and the nDCG@10 as well as nDCG and nDCG@10 when unjudged documents are removed (Cond. nDCG and Cond. nDCG@10)

Approach / Run	nDCG		nDCG@10		Cond. nDCG		Cond. nDCG@10	
	June	August	June	August	June	August	June	August
ows-bm25-bo1-keyqueries	**0.408**	**0.315**	**0.267**	**0.223**	**0.606**	**0.494**	**0.572**	**0.488**
ows-bm25-reverted-index	0.334	0.266	0.263	0.219	0.439	0.366	0.429	0.371
ows-ltr-all	0.305	0.242	0.224	0.175	0.426	0.352	0.414	0.355
ows-ltr-wows-rerank-and-reverted-index	0.301	0.244	0.219	0.187	0.432	0.361	0.424	0.364
ows-ltr-wows-rerank-and-keyquery	0.293	0.240	0.206	0.178	0.425	0.358	0.414	0.362
ows-ltr-wows-all-rerank	0.246	0.232	0.138	0.162	0.419	0.353	0.405	0.354
ows-ltr-wows-base-rerank	0.253	0.194	0.157	0.112	0.421	0.352	0.408	0.353
BM25	0.270	0.213	0.168	0.143	0.423	0.346	0.407	0.346

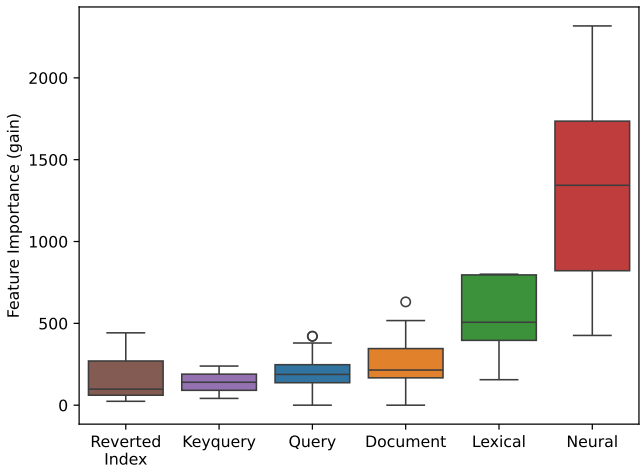

Figure 1: Feature importance per feature type. For feature importance, we use the 'gain' metric, which measures, per feature, the total performance gain obtained from splits using that feature.

the queries that were used before. Also, the approaches that use previously clicked documents perform much better compared to the approaches that do not use any historical information.

4.2. Learning to rank feature importance

Since our learning-to-rank approach uses a large number of different features, we were curious to see which features have the largest impact on the performance of the model. We inspect the 'gain' feature importance scores as reported by LightGBM [32], i.e., for each feature, the total gain obtained by splits in the decision tree in which that feature was used.

Figure 1 shows the feature importances per feature type. As can be expected, the query-document scores have the largest impact on the performance of the model, with the neural matching models being most important overall. Interestingly, the reverted index and keyquery features seem not to help the model all that much, even though we have seen large improvements in performance if we use those techniques directly (as opposed to only using them as features in the learning to rank model).

In Figure 2, we explore the 5 most important features for the query-only, document-only and query-document features. For the lexical matching models, we see that BM25 is the least important by a large margin. This could be caused by the fact we already use BM25 to select the top 100 documents before

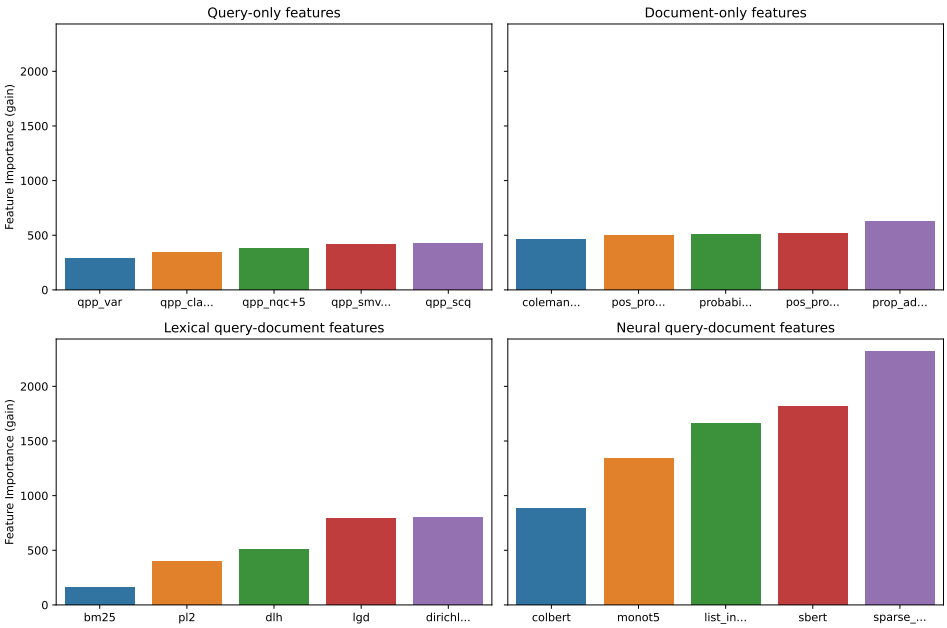

Figure 2: Top 5 most important features for query-only, document-only, and query-document (lexical and neural matching scores) features. For feature importance, we use the 'gain' metric, which measures, per feature, the total performance gain obtained from splits using that feature.

re-ranking, so the BM25 scores are already incorporated in the ranking. The neural ranking models, which were the most important features, still vary quite a bit in their importance. Interestingly, the sparse cross-encoder is weighed more heavily than models with full attention mechanisms like MonoT5, LiT5 and even more powerful models like RankZephyr. Similarly, SBERT, a bi-encoder model, is also deemed quite important by LightGBM. Importantly, this teaches us that we might not even need the most performant (e.g., full attention cross-encoder) models in our pipeline; using more lightweight models in a learning-to-rank setting might already boost performance by a large margin.

5. Conclusion

We presented the Open Web Search (OWS) team's submission to the LongEval shared task at CLEF 2024. The motivation behind our approach was twofold. For previously encountered queries, we made explicit use of the clicked documents in the past; either through a keyquery approach or by finding similar documents to the clicked documents in the new corpus. For unseen queries, we applied a learning-to-rank model with a variety of query-only, document-only and query-document features. Our results show that making explicit use of clicked documents for previously encountered queries heavily improves the performance of our system, even when the corpus has evolved in the meantime.

Acknowledgments

This work has received funding from the European Union's Horizon Europe research and innovation program under grant agreement No 101070014 (OpenWebSearch.EU, https://doi.org/10.3030/101070014).

References

[1] R. Alkhalifa, I. M. Bilal, H. Borkakoty, J. Camacho-Collados, R. Deveaud, A. El-Ebshihy, L. E. Anke, G. G. Sáez, P. Galuscáková, L. Goeuriot, E. Kochkina, M. Liakata, D. Loureiro, H. T. Madabushi, P. Mulhem, F. Piroi, M. Popel, C. Servan, A. Zubiaga, Longeval: Longitudinal evaluation of model

performance at CLEF 2023, in: J. Kamps, L. Goeuriot, F. Crestani, M. Maistro, H. Joho, B. Davis, C. Gurrin, U. Kruschwitz, A. Caputo (Eds.), Advances in Information Retrieval - 45th European Conference on Information Retrieval, ECIR 2023, Dublin, Ireland, April 2-6, 2023, Proceedings, Part III, volume 13982 of *Lecture Notes in Computer Science*, Springer, 2023, pp. 499–505. URL: https://doi.org/10.1007/978-3-031-28241-6_58. doi:10.1007/978-3-031-28241-6_58.

[2] R. Alkhalifa, I. M. Bilal, H. Borkakoty, J. Camacho-Collados, R. Deveaud, A. El-Ebshihy, L. E. Anke, G. N. G. Sáez, P. Galuscáková, L. Goeuriot, E. Kochkina, M. Liakata, D. Loureiro, P. Mulhem, F. Piroi, M. Popel, C. Servan, H. T. Madabushi, A. Zubiaga, Extended overview of the CLEF-2023 longeval lab on longitudinal evaluation of model performance, in: M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2023), Thessaloniki, Greece, September 18th to 21st, 2023, volume 3497 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 2181–2203. URL: https://ceur-ws.org/Vol-3497/paper-184.pdf.

[3] P. Galuscáková, R. Deveaud, G. G. Sáez, P. Mulhem, L. Goeuriot, F. Piroi, M. Popel, Longeval-retrieval: French-english dynamic test collection for continuous web search evaluation, in: H. Chen, W. E. Duh, H. Huang, M. P. Kato, J. Mothe, B. Poblete (Eds.), Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023, ACM, 2023, pp. 3086–3094. URL: https://doi.org/10.1145/3539618.3591921. doi:10.1145/3539618.3591921.

[4] R. Alkhalifa, H. Borkakoty, R. Deveaud, A. El-Ebshihy, L. E. Anke, T. Fink, G. G. Sáez, P. Galuscáková, L. Goeuriot, D. Iommi, M. Liakata, H. T. Madabushi, P. Medina-Alias, P. Mulhem, F. Piroi, M. Popel, C. Servan, A. Zubiaga, Longeval: Longitudinal evaluation of model performance at CLEF 2024, in: N. Goharian, N. Tonellotto, Y. He, A. Lipani, G. McDonald, C. Macdonald, I. Ounis (Eds.), Advances in Information Retrieval - 46th European Conference on Information Retrieval, ECIR 2024, Glasgow, UK, March 24-28, 2024, Proceedings, Part VI, volume 14613 of *Lecture Notes in Computer Science*, Springer, 2024, pp. 60–66. URL: https://doi.org/10.1007/978-3-031-56072-9_8. doi:10.1007/978-3-031-56072-9_8.

[5] R. Alkhalifa, H. Borkakoty, R. Deveaud, A. El-Ebshihy, L. E. Anke, T. Fink, G. G. Sáez, P. Galuscáková, L. Goeuriot, D. Iommi, M. Liakata, H. T. Madabushi, P. Medina-Alias, P. Mulhem, F. Piroi, M. Popel, C. Servan, A. Zubiaga, Overview of the CLEF-2023 LongEval Lab on Longitudinal Evaluation of Model Performance, in: G. F. N. Ferro, P. Galuščáková, A. G. S. de Herrera (Eds.), Working Notes of CLEF 2024 - Conference and Labs of the Evaluation Forum, CEUR-WS.org, 2024.

[6] R. Alkhalifa, H. Borkakoty, R. Deveaud, A. El-Ebshihy, L. Espinosa-Anke, T. Fink, P. Galuščáková, G. Gonzalez-Saez, L. Goeuriot, D. Iommi, M. Liakata, H. T. Madabushi, P. Medina-Alias, P. Mulhem, F. Piroi, M. Popel, A. Zubiaga, Overview of the CLEF 2024 LongEval Lab on Longitudinal Evaluation of Model Performance, in: L. Goeuriot, P. Mulhem, G. Quénot, D. Schwab, L. Soulier, G. M. D. Nunzio, P. Galuščáková, A. G. S. de Herrera, G. Faggioli, N. Ferro (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fifteenth International Conference of the CLEF Association (CLEF 2024), Lecture Notes in Computer Science (LNCS), Springer, Heidelberg, Germany, 2024.

[7] R. Alkhalifa, H. Borkakoty, R. Deveaud, A. El-Ebshihy, L. Espinosa-Anke, T. Fink, P. Galuščáková, G. Gonzalez-Saez, L. Goeuriot, D. Iommi, M. Liakata, H. T. Madabushi, P. Medina-Alias, P. Mulhem, F. Piroi, M. Popel, A. Zubiaga, Extended overview of the CLEF 2024 LongEval Lab on Longitudinal Evaluation of Model Performance, in: G. Faggioli, N. Ferro, P. Galuščáková, A. G. S. de Herrera (Eds.), Working Notes of CLEF 2024 - Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS, Online, 2024.

[8] T. Gollub, M. Hagen, M. Michel, B. Stein, From Keywords to Keyqueries: Content Descriptors for the Web, in: C. Gurrin, G. Jones, D. Kelly, U. Kruschwitz, M. de Rijke, T. Sakai, P. Sheridan (Eds.), 36th International ACM Conference on Research and Development in Information Retrieval (SIGIR 2013), ACM, 2013, pp. 981–984. doi:10.1145/2484028.2484181.

[9] M. Hagen, A. Beyer, T. Gollub, K. Komlossy, B. Stein, Supporting Scholarly Search with Keyqueries, in: N. Ferro, F. Crestani, M.-F. Moens, J. Mothe, F. Silvestri, G. Di Nunzio, C. Hauff, G. Silvello (Eds.), Advances in Information Retrieval. 38th European Conference on IR Research (ECIR 2016),

volume 9626 of *Lecture Notes in Computer Science*, Springer, Berlin Heidelberg New York, 2016, pp. 507–520. doi:10.1007/978-3-319-30671-1_37.

[10] S. MacAvaney, N. Tonellotto, C. Macdonald, Adaptive re-ranking with a corpus graph, in: M. A. Hasan, L. Xiong (Eds.), Proceedings of the 31st ACM International Conference on Information & Knowledge Management, Atlanta, GA, USA, October 17-21, 2022, ACM, 2022, pp. 1491–1500. URL: https://doi.org/10.1145/3511808.3557231. doi:10.1145/3511808.3557231.

[11] J. Pickens, M. Cooper, G. Golovchinsky, Reverted indexing for feedback and expansion, in: J. X. Huang, N. Koudas, G. J. F. Jones, X. Wu, K. Collins-Thompson, A. An (Eds.), Proceedings of the 19th ACM Conference on Information and Knowledge Management, CIKM 2010, Toronto, Ontario, Canada, October 26-30, 2010, ACM, 2010, pp. 1049–1058. URL: https://doi.org/10.1145/1871437.1871571. doi:10.1145/1871437.1871571.

[12] T. Liu, Learning to Rank for Information Retrieval, Springer, 2011. URL: https://doi.org/10.1007/978-3-642-14267-3. doi:10.1007/978-3-642-14267-3.

[13] J. Lin, R. F. Nogueira, A. Yates, Pretrained Transformers for Text Ranking: BERT and Beyond, Synthesis Lectures on Human Language Technologies, Morgan & Claypool Publishers, 2021. URL: https://doi.org/10.2200/S01123ED1V01Y202108HLT053. doi:10.2200/S01123ED1V01Y202108HLT053.

[14] D. Dato, S. MacAvaney, F. M. Nardini, R. Perego, N. Tonellotto, The istella22 dataset: Bridging traditional and neural learning to rank evaluation, in: E. Amigó, P. Castells, J. Gonzalo, B. Carterette, J. S. Culpepper, G. Kazai (Eds.), SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022, ACM, 2022, pp. 3099–3107. URL: https://doi.org/10.1145/3477495.3531740. doi:10.1145/3477495.3531740.

[15] M. Fröbe, S. Günther, M. Probst, M. Potthast, M. Hagen, The Power of Anchor Text in the Neural Retrieval Era, in: M. Hagen, S. Verberne, C. Macdonald, C. Seifert, K. Balog, K. Nørvåg, V. Setty (Eds.), Advances in Information Retrieval. 44th European Conference on IR Research (ECIR 2022), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2022.

[16] S. M. Farzana, M. Fröbe, M. Granitzer, G. Hendriksen, D. Hiemstra, M. Potthast, S. Zerhoudi, The first international workshop on open web search (wows), in: European Conference on Information Retrieval, Springer, 2024, pp. 426–431.

[17] S. M. Farzana, M. Fröbe, M. Granitzer, G. Hendriksen, D. Hiemstra, M. Potthast, S. Zerhoudi (Eds.), Proceedings of the first International Workshop on Open Web Search co-located with the 46th European Conference on Information Retrieval ECIR 2024, number 3689 in CEUR Workshop Proceedings, 2024. URL: https://ceur-ws.org/Vol-3689/.

[18] K. Krishna, A. Roy, M. Iyyer, Hurdles to progress in long-form question answering, in: K. Toutanova, A. Rumshisky, L. Zettlemoyer, D. Hakkani-Tür, I. Beltagy, S. Bethard, R. Cotterell, T. Chakraborty, Y. Zhou (Eds.), Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021, Association for Computational Linguistics, 2021, pp. 4940–4957. URL: https://doi.org/10.18653/v1/2021.naacl-main.393. doi:10.18653/V1/2021.NAACL-MAIN.393.

[19] M. Fröbe, C. Akiki, M. Potthast, M. Hagen, How Train-Test Leakage Affects Zero-shot Retrieval, in: D. Arroyuelo, B. Poblete (Eds.), 29th International Symposium on String Processing and Information Retrieval (SPIRE 2022), volume 13617, Concepción, Chile, 2022. doi:10.1007/978-3-031-20643-6_11.

[20] Y. Bernstein, J. Zobel, Redundant documents and search effectiveness, in: O. Herzog, H. Schek, N. Fuhr, A. Chowdhury, W. Teiken (Eds.), Proceedings of the 2005 ACM CIKM International Conference on Information and Knowledge Management, Bremen, Germany, October 31 - November 5, 2005, ACM, 2005, pp. 736–743. URL: https://doi.org/10.1145/1099554.1099733. doi:10.1145/1099554.1099733.

[21] M. Fröbe, J. Bittner, M. Potthast, M. Hagen, The Effect of Content-Equivalent Near-Duplicates on the Evaluation of Search Engines, in: J. Jose, E. Yilmaz, J. Magalhães, P. Castells, N. Ferro, M. Silva, F. Martins (Eds.), Advances in Information Retrieval. 42nd European Conference on IR Research (ECIR 2020), volume 12036 of *Lecture Notes in Computer Science*, Springer, Berlin Heidelberg New

York, 2020, pp. 12–19. doi:10.1007/978-3-030-45442-5_2.
[22] M. Fröbe, J. Bevendorff, J. Reimer, M. Potthast, M. Hagen, Sampling Bias Due to Near-Duplicates in Learning to Rank, in: 43rd International ACM Conference on Research and Development in Information Retrieval (SIGIR 2020), ACM, 2020, pp. 1997–2000. doi:10.1145/3397271.3401212.
[23] M. Fröbe, J. Bevendorff, L. Gienapp, M. Völske, B. Stein, M. Potthast, M. Hagen, CopyCat: Near-Duplicates within and between the ClueWeb and the Common Crawl, in: F. Diaz, C. Shah, T. Suel, P. Castells, R. Jones, T. Sakai (Eds.), 44th International ACM Conference on Research and Development in Information Retrieval (SIGIR 2021), ACM, 2021, pp. 2398–2404. doi:10.1145/3404835.3463246.
[24] M. Völske, T. Gollub, M. Hagen, B. Stein, A keyquery-based classification system for CORE, D Lib Mag. 20 (2014). URL: https://doi.org/10.1045/november14-voelske. doi:10.1045/NOVEMBER14-VOELSKE.
[25] M. Fröbe, S. Günther, A. Bondarenko, J. Huck, M. Hagen, Using keyqueries to reduce misinformation in health-related search results, in: ROMCIR 2022: The 2nd Workshop on Reducing Online Misinformation through Credible Information Retrieval, held as part of ECIR 2022: the 44th European Conference on Information Retrieval, 2022.
[26] M. Fröbe, E. O. Schmidt, M. Hagen, Efficient Query Obfuscation with Keyqueries, in: 20th International IEEE/WIC/ACM Conference on Web Intelligence (WI-IAT 2021), ACM, 2021. doi:10.1145/3486622.3493950.
[27] M. Fröbe, G. Hendriksen, A. P. de Vries, M. Potthast, Open web search at longeval 2023: Reciprocal rank fusion on automatically generated query variants, in: M. Aliannejadi, G. Faggioli, N. Ferro, M. Vlachos (Eds.), Working Notes of the Conference and Labs of the Evaluation Forum (CLEF 2023), Thessaloniki, Greece, September 18th to 21st, 2023, volume 3497 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2023, pp. 2432–2440. URL: https://ceur-ws.org/Vol-3497/paper-195.pdf.
[28] G. Amati, Divergence from randomness, Ph.D. thesis, Ph. D. thesis, Department of Computer Science, University of Glasgow, 2003.
[29] N. A. Jaleel, J. Allan, W. B. Croft, F. Diaz, L. S. Larkey, X. Li, M. D. Smucker, C. Wade, Umass at TREC 2004: Novelty and HARD, in: E. M. Voorhees, L. P. Buckland (Eds.), Proceedings of the Thirteenth Text REtrieval Conference, TREC 2004, Gaithersburg, Maryland, USA, November 16-19, 2004, volume 500-261 of *NIST Special Publication*, National Institute of Standards and Technology (NIST), 2004. URL: http://trec.nist.gov/pubs/trec13/papers/umass.novelty.hard.pdf.
[30] Q. Wu, C. J. Burges, K. M. Svore, J. Gao, Adapting boosting for information retrieval measures, Information Retrieval 13 (2010) 254–270.
[31] C. Macdonald, N. Tonellotto, Declarative experimentation in information retrieval using pyterrier, in: Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval, 2020, pp. 161–168.
[32] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, LightGBM: A Highly Efficient Gradient Boosting Decision Tree, Advances in Neural Information Processing Systems 30 (2017).
[33] M. Fröbe, J. H. Reimer, S. MacAvaney, N. Deckers, S. Reich, J. Bevendorff, B. Stein, M. Hagen, M. Potthast, The information retrieval experiment platform, in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2023, pp. 2826–2836.
[34] O. Zendel, M. Fröbe, G. Faggioli, Qpptk@ tirex: Simplified query performance prediction for ad-hoc retrieval experiments, in: [17], 2024, pp. 50–62. URL: https://ceur-ws.org/Vol-3689/.
[35] D. Alexander, W. Kusa, A. P. de Vries, Orcas-i query intent predictor as component of tira, in: [17], 2024, pp. 23–29. URL: https://ceur-ws.org/Vol-3689/.
[36] F. Schlatt, Efficiently scoring the health-relatedness of web pages, in: [17] 14–22. URL: https://ceur-ws.org/Vol-3689/.
[37] L. Erben, M. Hampel, M.-C. Kuns, V. Melisch, P. Natzschka, W. Pertsch, L. Razouk, R. Stolle, R. T. Thoss, T. G. Trinh, et al., Assembling four open web search components, in: [17] 73–93. URL: https://ceur-ws.org/Vol-3689/.
[38] R. Pradeep, S. Sharifymoghaddam, J. Lin, Rankzephyr: Effective and robust zero-shot listwise

reranking is a breeze!, arXiv preprint arXiv:2312.02724 (2023).
[39] F. Schlatt, M. Fröbe, M. Hagen, Investigating the effects of sparse attention on cross-encoders, in: European Conference on Information Retrieval, Springer, 2024, pp. 173–190.
[40] M. S. Tamber, R. Pradeep, J. Lin, Scaling down, litting up: Efficient zero-shot listwise reranking with seq2seq encoder-decoder models, arXiv preprint arXiv: 2312.16098 (2023).
[41] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2019. URL: https://arxiv.org/abs/1908.10084.
[42] R. Nogueira, Z. Jiang, R. Pradeep, J. Lin, Document ranking with a pretrained sequence-to-sequence model, in: T. Cohn, Y. He, Y. Liu (Eds.), Findings of the Association for Computational Linguistics: EMNLP 2020, Association for Computational Linguistics, Online, 2020, pp. 708–718. URL: https://aclanthology.org/2020.findings-emnlp.63. doi:10.18653/v1/2020.findings-emnlp.63.
[43] O. Khattab, M. Zaharia, Colbert: Efficient and effective passage search via contextualized late interaction over bert, in: Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, 2020, pp. 39–48.
[44] L. Xiong, C. Xiong, Y. Li, K.-F. Tang, J. Liu, P. Bennett, J. Ahmed, A. Overwijk, Approximate nearest neighbor negative contrastive learning for dense text retrieval, arXiv preprint arXiv:2007.00808 (2020).
[45] K. Järvelin, J. Kekäläinen, Cumulated gain-based evaluation of IR techniques, ACM Trans. Inf. Syst. 20 (2002) 422–446. URL: http://doi.acm.org/10.1145/582415.582418. doi:10.1145/582415.582418.
[46] T. Sakai, Alternatives to bpref, in: W. Kraaij, A. P. de Vries, C. L. A. Clarke, N. Fuhr, N. Kando (Eds.), SIGIR 2007: Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Amsterdam, The Netherlands, July 23-27, 2007, ACM, 2007, pp. 71–78. URL: https://doi.org/10.1145/1277741.1277756. doi:10.1145/1277741.1277756.
[47] T. Sakai, Comparing metrics across TREC and NTCIR: the robustness to system bias, in: J. G. Shanahan, S. Amer-Yahia, I. Manolescu, Y. Zhang, D. A. Evans, A. Kolcz, K. Choi, A. Chowdhury (Eds.), Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 2008, Napa Valley, California, USA, October 26-30, 2008, ACM, 2008, pp. 581–590. URL: https://doi.org/10.1145/1458082.1458159. doi:10.1145/1458082.1458159.
[48] M. Fröbe, L. Gienapp, M. Potthast, M. Hagen, Bootstrapped nDCG Estimation in the Presence of Unjudged Documents, in: Advances in Information Retrieval. 45th European Conference on IR Research (ECIR 2023), volume 13980 of *Lecture Notes in Computer Science*, Springer, Berlin Heidelberg New York, 2023, pp. 313–329. doi:10.1007/978-3-031-28244-7_20.