# The Power of Anchor Text in the Neural Retrieval Era

Maik Fröbe,[1] Sebastian Günther,[1] Maximilian Probst,[1]
Martin Potthast,[2] Matthias Hagen[1]

[1] Martin-Luther-Universität Halle-Wittenberg
[2] Leipzig University

**Abstract** In the early days of web search, a study by Craswell et al. [11] showed that anchor texts are particularly helpful ranking features for navigational queries and a study by Eiron and McCurley [24] showed that anchor texts closely resemble the characteristics of queries and that retrieval against anchor texts yields more homogeneous results than against documents. In this reproducibility study, we analyze to what extent these observations still hold in the web search scenario of the current MS MARCO dataset, including the paradigm shift caused by pre-trained transformers. Our results show that anchor texts still are particularly helpful for navigational queries, but also that they only very roughly resemble the characteristics of queries and that they now yield less homogeneous results than the content of documents. As for retrieval effectiveness, we also evaluate anchor text from different time frames and include modern baselines in a comparison on the TREC 2019 and 2020 Deep Learning tracks. Our code and the newly created large-scale anchor text datasets for the MS MARCO dataset are freely available.

**Keywords:** Anchor text · MS MARCO · ORCAS · TREC Deep Learning track

## 1 Introduction

Almost from the beginning, search engines have exploited the web's link structure to improve their result rankings. But besides the actual links, also the anchor texts (i.e., the clickable texts of the links) were an important ranking feature, since they "often provide more accurate descriptions of web pages than the pages themselves" [2].

The seminal works of Craswell et al. [11] and Eiron and McCurley [24] from 2001 and 2003 examined two important aspects of anchor text. Craswell et al. showed that anchor text especially helps for navigational queries (i.e., queries to find a specific document [3]). This result explained why commercial search engines heavily used anchor text even though no positive effect was observed in TREC scenarios [27, 47]: more than 20% of the traffic of commercial search engines were navigational queries [3], but hardly any TREC topic was navigational. Eiron and McCurley showed that retrieval against anchor texts yields more homogeneous results than against documents and that anchor texts closely resemble the characteristics of queries. This result later inspired others to use anchor texts as a replacement for proprietary query logs [7, 20, 36, 38].

In the two decades since the studies of Craswell et al. and Eiron and McCurley were published, the web and the search behavior of users have changed. We thus analyze to what extent the original findings can be reproduced on current web crawls and query

logs. Additionally, given the recent success of pre-trained transformers [50], we also analyze whether anchor text is still a valuable ranking feature or whether it might be "obsolete" for retrieval pipelines using BERT [42], MonoT5 [43], or DeepCT [18].

As the scenario for our reproducibility study, we employ the MS MARCO dataset (3.2 and 12 million documents, 367,013 queries with relevance judgments) [15], the ORCAS query log (18.8 million query-click entries related to MS MARCO documents) [8], and extract anchor texts from Common Crawl snapshots of the last six years. Our new anchor text datasets contain billions of anchor texts for about 1.7 million documents from MS MARCO version 1 (about 53% of all documents) and for about 4.82 million documents from MS MARCO version 2 (about 40% of all documents).

The results of our reproducibility study are dichotomous. While we can reproduce Craswell et al.'s observation that anchor text is particularly helpful for navigational queries (details in Section 5), we find substantial differences for the results of Eiron and McCurley. In the MS MARCO scenario, the anchor texts are pretty different to queries (e.g., number of distinct terms) and retrieval against them yields less (not more) homogeneous results than against the content of documents (details in Section 4). We attribute both changes to the fact that Eiron and McCurley conducted their study in the corporate IBM intranet with queries and anchor texts both formulated by employees of IBM, whereas, in our reproducibility scenario, we have "arbitrary" searchers and anchor text authors from the web. In the reproducibility experiments for the study of Craswell et al., we also evaluate the effectiveness of anchor text from different time frames and include modern baselines in a comparison on the topics of the TREC 2019 and 2020 Deep Learning tracks. The results still confirm the observation that anchor text only slightly improves the effectiveness in TREC scenarios [11, 27, 47]. All our code and data is published under a permissible open-source license.[3]

## 2 Related Work

Exploiting link structure has a long tradition in IR [16]. Already in 1993, Dunlop and van Rijsbergen [23] used text referring to non-textual objects like images to retrieve those non-textual objects for text queries. McBryan [40] refined this process by only including terms from the clickable texts of links: the *anchor texts*. Anchor texts were later reported to be heavily used by commercial search engines [2, 24] but had no positive effect in TREC scenarios [1, 26, 27, 47]. Craswell et al. [11] resolved this dichotomy by showing that anchor text is particularly useful for navigational queries (i.e., queries to find a specific document [3]) while hardly any TREC topics were navigational.

After Craswell et al.'s result, dedicated shared tasks like homepage finding or named page finding evolved [9, 12, 10] and more and more systems incorporated anchor text for navigational queries. For instance, Westerveld et al. [47] combined anchor text with a document's content, URL, and link count and Ogilvie and Callan [44] showed that anchor text can also be combined with poor performing features without harming the overall effectiveness for navigational queries. Since links may "rot" over time [34]—resulting in possibly outdated anchor texts—, several approaches used historical information [17] or importance estimation [22, 41] to weight anchor text. Finally, the anchor

[3] Code and data: https://github.com/webis-de/ecir22-anchor-text

text source and quantity were shown to be very important. Kamps et al. [29] found that anchor text from the Wikipedia is more effective than anchor text from the general web while Koolen and Kamps [35] showed that more anchor text led to higher early precision on the TREC 2009 Web track [6] including 66 navigational subtopics.

Anchor text became an important retrieval feature and also served as a replacement for query logs [24, 20, 36, 7, 38]. But with the recent paradigm shift caused by transformers [50], large parts of the IR community moved from feature engineering to neural re-rankers and dense retrieval models [30], e.g., on the MS MARCO datasets in the TREC Deep Learning tracks [14, 8]. In this new context, we aim to reproduce the seminal anchor text studies by Craswell et al. [11] and Eiron and McCurley [24].

## 3 Anchor Text Dataset for MS MARCO

There is no publicly available anchor text dataset for MS MARCO and the documents themselves are only sparsely linked. To still reproduce the results of Craswell et al. and Eiron and McCurley on MS MARCO, we extract anchor texts from Common Crawl snapshots. We randomly select one snapshot from each year from 2016 to 2021 (each containing 1.7–3.4 billion documents) and extract the anchor texts of links to MS MARCO documents. However, different to the studies of Craswell et al. and Eiron and McCurley, we do not simply keep all anchor texts but apply several natural filtering steps that were also used previously to remove lower-quality anchor texts [5]. First, we ignore anchor texts consisting only of anchor text "stop words" (manually selected: click, read, link, mail, here, open). Second, we ignore anchor texts with more than 10 words since these often resulted from parsing errors. And third, we ignore intra-site anchor texts (i.e., with the same source and target domain) since anchor text of inter-site links is usually more descriptive [41]. These filtering steps remove about 50% of all anchor texts pointing to MS MARCO documents.

On our Hadoop cluster with 3000 CPU cores, we processed 17.12 billion documents (343 TiB of compressed WARC files) and extracted 8.16 billion anchor texts for MS MARCO documents. In a first data analysis, we observed that most of the anchor texts point to only a few very popular documents. To keep the dataset size feasible for our planned experiments, we decided to min-wise sample 1,000 anchor texts for documents with more than 1,000 anchor texts. Note that this stratified sampling still ensures that we include all anchor texts for most of the documents (94% for MS MARCO version 1; 97% for version 2) while we downsample for the most popular documents.

Table 1 shows an overview of all extracted anchor texts (column group 'Anchors') and the downsampled subsets for the two MS MARCO versions ('Sample@V1' and 'Sample@V2'). Overall, the combined samples cover 1.70 million documents of version 1 (53 % of all documents) and 4.82 million documents of version 2 (40 %). For each anchor text, our datasets also contain the source URL, the target URL, and the MS MARCO ID of the target document. We use these datasets in our reproducibility study of the main findings of Eiron and McCurley [24] (similarity of anchor text and queries; cf. Section 4) and of the retrieval effectiveness results of Craswell et al. [11] (anchor text particularly helps for navigational queries; cf. Section 5).

**Table 1.** Overview of our anchor text datasets for MS MARCO. The samples for version 1 and 2 (Sample@V1 / V2) include 1,000 anchor texts min-wise sampled per target document.

| Common Crawl Snapshot | | | Anchors | | Sample@V1 | | Sample@V2 | |
|---|---|---|---|---|---|---|---|---|
| | Docs | Size | V1 | V2 | Anchors | Docs Cov. | Anchors | Docs Cov. |
| 2016-07 | 1.73 b | 28.57 TiB | 1.05 b | 0.75 b | 54.05 m | 0.83 m | 65.04 m | 1.49 m |
| 2017-04 | 3.14 b | 53.95 TiB | 0.95 b | 0.91 b | 61.19 m | 1.18 m | 94.35 m | 2.34 m |
| 2018-13 | 3.20 b | 67.66 TiB | 0.83 b | 0.68 b | 81.24 m | 1.27 m | 116.59 m | 2.45 m |
| 2019-47 | 2.55 b | 53.95 TiB | 0.55 b | 0.41 b | 65.60 m | 1.16 m | 90.18 m | 2.83 m |
| 2020-05 | 3.10 b | 59.94 TiB | 0.67 b | 0.48 b | 78.46 m | 1.24 m | 108.16 m | 3.10 m |
| 2021-04 | 3.40 b | 78.98 TiB | 0.52 b | 0.36 b | 60.62 m | 1.14 m | 84.93 m | 3.18 m |
| $\sum$ | 17.12 b | 343.05 TiB | 4.57 b | 3.59 b | 207.28 m | 1.70 m | 341.17 m | 4.82 m |

## 4 Properties of Anchor Texts, Queries, and Documents

In 2003, Eiron and McCurley [24] studied properties of anchor texts, queries, and documents on an IBM intranet (2.95 million documents, 2.57 million anchor texts, and 1.27 million queries). They found that anchor texts closely resembled query length, that terms in document titles/bodies and in anchor texts often have different meanings, and that retrieval against anchor text yielded more homogeneous results than against document content. Eiron and McCurley also conducted a study on retrieval effectiveness but we do not reproduce their setup without relevance judgments but instead reproduce the retrieval experiments of Craswell et al. [11] with relevance judgments (cf. Section 5).

Analyzing to what extent the similarity of anchor texts and queries that Eiron and McCurley observed can be reproduced in a current retrieval scenario is particularly important since the observation had inspired others to replace proprietary query logs by anchor texts [7, 20, 38]. We repeat the study of Eiron and McCurley on the MS MARCO version 1 dataset and the ORCAS query log [8] linked to it. Interestingly, in our "modern" web search scenario with about 27 times more anchor texts (81.24 million in the 2018 subset matching the MS MARCO version 1 crawling date) and 15 times more queries (18.82 million from ORCAS), we obtain some substantially different results.

**Number of Distinct Terms.** The left plots in Figure 1 show the distributions of the number of distinct terms per anchor text, query, or document title as reported by Eiron and McCurley for their IBM dataset (upper plot) and what we observe for MS MARCO (lower plot). While Eiron and McCurley reported the distributions for anchor texts and queries as highly similar, we find them to be rather dissimilar on MS MARCO.

To assess the similarity of the distributions, we calculate the symmetric Jensen-Shannon distance [25] for all pairs (right plot of Figure 1; a distance of 0 indicates equal distributions). The anchor text distributions are very similar for the MS MARCO and the IBM data (distance of 0.099) as are the distributions of anchor texts and queries for the IBM data (0.14). However, on the MS MARCO data, anchor texts and queries are more dissimilar (0.28) probably mainly due to the more "web-like" query distribution: the IBM query distribution is pretty different to the ORCAS queries (distance of 0.34; most IBM queries have one term, most ORCAS queries have three terms, etc.).

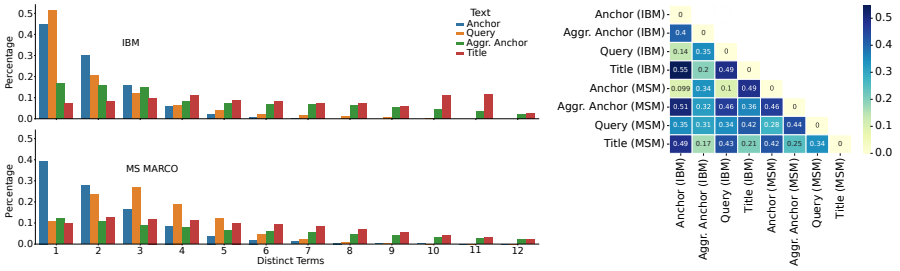



**Figure 1.** (Left) Distributions of the number of distinct terms in anchor texts, queries, document titles, and aggregated anchor texts (all anchors combined that point to a document) on the IBM data and MS MARCO. (Right) Jensen-Shannon distance of all pairs (0 means identical).

**Frequent Terms.** Eiron and McCurley also compared the 16 most frequent terms in document titles, queries, and anchor texts and found that these are rather different. Some terms like 'of' are frequent in all types but most terms frequent in one type are rare in the other types. Eiron and McCurley then argued that the different frequencies indicate that anchor texts should be kept separate and not mixed with document content such that methods depending on term frequencies could better exploit the different contexts of a term's frequencies. We can confirm the observed substantial differences also for the MS MARCO scenario. For example, the frequent terms 'you', 'it', and 'are' for titles, 'meaning', 'online', and 'free' for ORCAS queries, as well as 'home', 'university', or 'website' for anchor texts very rarely occur in the other types.

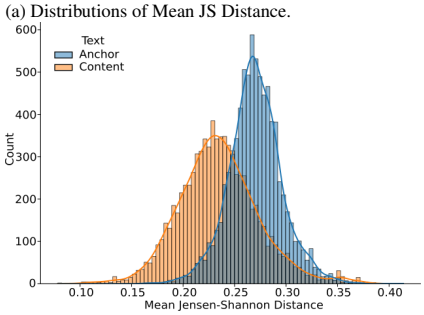
**Search Result Homogeneity.** Eiron and McCurley reported that most of the queries in their log were navigational (e.g., `benefits` or `travel` to find respective IBM guidelines) and that matching queries in the document content tended to retrieve results for every possible meaning of the query terms while matching only in the anchor texts retrieved more homogeneous results—but in an experiment with only 14 queries.

On 10,000 randomly sampled ORCAS queries, we follow the setup of Eiron and McCurley: we rank the MS MARCO documents by either matching their anchor texts or their content, we remove queries with less than 800 results (7,962 queries remain), and we measure the results' homogeneity using the method of Kilgarriff and Rose [33] to compute the mean Jensen-Shannon distances; distributions shown in Table 2 (a).

In contrast to Eiron and McCurley, we observe that retrieval against document content yields more homogeneous results than against anchor text (cf. Table 2 (c); content yields more homogeneous results for more than 6,700 queries). For example, the top-100 content-based results for the query `kelly clarkson fan club` all refer to Kelly Clarkson while the anchor text-based results are more "diverse" (cf. the excerpts in Table 2 (b)). An explanation for the difference to the observation of Eiron and McCurley probably is twofold: (1) our large-scale dataset has rather diverse authors and queries from different searchers while in the IBM data anchor text writers and searchers probably were IBM employees with experience in intranet search, and, probably more importantly, (2) Eiron and McCurley have experimented with 14 queries only.

**Table 2.** Homogeneity of anchor text- and content-based search results: (a) mean Jensen-Shannon distance, (b) result excerpts for query with largest distance, (c) number of queries with significantly more (↑) or less (↓) homogeneous content-based results.

(a) Distributions of Mean JS Distance.




(b) Ranking excerpts for query `kelly clarkson fan club`.

| **Index:** Anchor | **Mean JS Distance:** 0.32 |
|---|---|
| **Rank** | **Page Description** |
| 1 | Homepage of Kelly Clarkson |
| 2 | Wikipedia article about Kelly Clarkson |
| 3 | Overview of Latest News on Kelly Clarkson |
| ⋮ | |
| 98 | Statistics on the american movie icon Grace Kelly |
| 99 | Job openings page by company kelly services |
| 100 | Login Page to facebook |
| **Index:** Content | **Mean JS Distance:** 0.13 |
| 1 | News article about Kelly Clarkson's career |
| 2 | Wikipedia article about a single by Kelly Clarkson |
| 3 | IMDb biography of Kelly Clarkson |
| ⋮ | |
| 98 | News article about released album by Kelly Clarkson |
| 99 | Review of Kelly Clarkson's show at American Idol |
| 100 | Article about weight gain of Kelly Clarkson |

(c) Significant Differences.

| p-value | ↑ | ↓ | ↔ |
|---|---|---|---|
| 0.05 | 6,770 | 1,121 | 71 |
| 0.01 | 6,764 | 1,113 | 85 |
| 0.001 | 6,748 | 1,098 | 116 |

## 5 Anchor Text and Retrieval Effectiveness

We compare the retrieval effectiveness against the MS MARCO anchor text datasets with traditional and modern content-based retrieval models for navigational queries to reproduce the result of Craswell et al. [11] (that anchor text particularly helps for navigational queries) but we also further extend the experiment to informational queries from the TREC Deep Learning tracks [14, 13, 15].

### 5.1 Navigational Queries for MS Marco

Craswell et al. [11] constructed 300 navigational queries split into three parts to demonstrate the effectiveness of anchor text for web search. In a web crawl with 18.5 million documents, they constructed 100 navigational queries pointing to random entry pages and 100 navigational queries pointing to popular entry pages randomly selected from Yahoo! (i.e., a manually maintained list of popular entry pages). Additionally, they constructed 100 navigational queries pointing to academic persons or institutions in a crawl of 0.4 million documents from the domain of the Australian National University (we omit those academic queries from our reproduction to focus on general web search).

We construct 200 navigational topics following the methodology of Craswell et al. [11]. Therefore, we extract all MS MARCO documents that potentially are entry pages based on their URLs using the rules by Westerveld et al. [47] that the URL-path of documents must be equal to index.html or the empty string, providing us with 92,562 candidates. From our entry page candidates, we selected 100 documents at random to construct queries pointing to random entry pages. To obtain queries pointing at popular entry pages, we selected 100 documents at random from the entry page candidates whose domains are listed in the Alexa top-1000 ranking of 2018. Finally, we construct

our 200 navigational queries by manually inspecting the selected target document and formulating a query that searchers would likely use for this target page. Due to our topic construction, we can not include retrieval algorithms into our evaluation that use the URL or the Alexa rank as features because we used both for the topic construction.

We transfer our topics that we have created for MS MARCO version 1 to MS MARCO version 2 by identifying whether the URL is present in MS MARCO version 2 and checking that the page has not changed, i.e., that the same navigational query still applies. With this approach, we can directly reuse models trained on MS MARCO version 1 on version 2 without risking train test leakage or train models anew.

### 5.2 Retrieval Models and Training

We use 18 retrieval models to reproduce the work of Craswell et al. [11] on the effectiveness of anchor text for navigational queries, inspecting novel retrieval models, covering six years of anchor text. Seven of those models use only the anchor text crawled at different points in time to create rankings with BM25. From our 11 models for comparison, six solely use the document's content, while the remaining five use combinations of the ORCAS query log [8], our anchor text, and the document's title and body.

All but two of the 11 retrieval models that we use for comparison against anchor text employ novel approaches (DeepCT [18, 19], MonoBERT [42], MonoT5 [43], and LambdaMART [4]) that did not exist during the evaluation of Craswell et al. [11]. Especially DeepCT, which predicts the importance of terms in their context, is an interesting novel approach because it can use traditional TREC-style relevance labels, query logs, or anchor text as its term-importance scores for training. Hence, we train three different DeepCT models and compare the term importance scores assigned by those three models as an additional case study for the similarity of anchor text to query logs. Altogether, our setup expands the work by Craswell et al. [11] who compared two retrieval models (BM25 on the content vs. BM25 on the anchor text) with novel aspects because we include modern models and evaluate the effectiveness of anchor text over time.

We use the Anserini toolkit [49] to implement our retrieval experiments. Following Craswell et al. [11], we do not tune the parameters of BM25 for our experiments, keeping them at Anserini's defaults (k=0.9 and b=0.4). We preprocess queries and the indexed text by stemming with the porter stemmer and stopword removal using Lucene's default stopwords for English. For re-ranking documents using MonoT5 and MonoBERT, we follow Nogueira et al. [43] and omit stemming and stopword removal. For all rankers, We break score ties within runs using alphanumeric ordering by document ID implemented in Anserini [49] (given random document IDs, this leads to a random distribution regarding other document properties such as the text length [37]).

**BM25 on Anchor Text.** Following Craswell et al. [11], we concatenate all anchor texts pointing to the same target page and index only this aggregated anchor text. We create dedicated Anserini indexes [49] for all 14 anchor text samples (see Table 1). With this anchor text retrieval, we mimic the corresponding baseline of Craswell et al. [11] with the novel aspect that we have multiple indexes covering six years of anchor text.

**BM25 on Content.** Mimicking the corresponding baseline by Craswell et al. [11], we concatenate the title and body of the documents and index them with Anserini [49].

**Table 3.** (a) Overview of the term importance datasets for DeepCT. (b) Pairwise comparison of term importance datasets on MS MARCO in terms of correlations (Kendall's $\tau$, Pearson's $\rho$) and Jaccard similarity ($J$) according to the ORCAS query log (ORCAS), the official training data (Train), and our extracted anchor text from the Common Crawl union 2016–2021 (Anchor).

(a) Overview of our Term-Importance Datasets.

| Term Importance Dataset | Docs | Passages | Empty |
|---|---|---|---|
| Anchor | 1.43 m | 11.64 m | 2.02 m |
| ORCAS | 0.88 m | 8.17 m | 0.92 m |
| Train | 0.25 m | 2.08 m | 0.29 m |

(b) Comparison of Importance Scores.

| | $\tau$ | $\rho$ | $J$ |
|---|---|---|---|
| ORCAS vs. Anchor | **0.39** | **0.61** | **0.53** |
| ORCAS vs. Train | 0.35 | 0.46 | 0.51 |
| Anchor vs. Train | 0.26 | 0.41 | 0.45 |

**DeepCT on Content.** DeepCT [18, 19] estimates the importance of terms in their context, removing unimportant terms while including multiple copies of important terms. Therefore, DeepCT is precision-oriented, making it a promising novel baseline for navigational queries. We train three DeepCT models on the official relevance judgments of version 1 of MS MARCO, the ORCAS query log, and our new anchor text.

We follow Dai and Callan [18] and use the fraction of queries respectively anchor texts of a document containing a term as the importance of that term in the document. Hence, we construct three datasets, obtaining the term importance for training from (1) queries in the official training data, (2) queries in the ORCAS query log, and (3) anchor text in our new anchor text sample. We ensure that our training data has no train/test leakage to our navigational topics by removing all documents with queries or anchor text having a term from our navigational topics. This filtering step identifies 270,511 documents that might cause a train/test leakage that we remove from all three datasets, providing us with 1,432,621 training documents for anchor text, 249,046 training documents for the official training data, and 876,950 training documents for the ORCAS query log (see Table 3a for a detailed overview of our three datasets).

Our three training datasets allow us to compare the term importance scores assigned by queries to those given by anchor texts as novel aspect to study their similarity. Table 3b shows a pairwise comparison. We report the Jaccard similarity of terms with non-zero importance and the correlation between the term-importance scores as Kendalls $\tau$ and Pearsons $\rho$. All our reported measures show that the ORCAS query log and our anchor text is the combination with the highest similar term importance scores.

We train our three DeepCT models with the implementation by Dai and Callan [18], process documents using PyTerrier [39], and index the processed documents with Anserini [49]. We follow the suggestions by Dai and Callan [19] and train each DeepCT model with a maximum input length of 512 tokens for 100 k steps with a batch size of 16 and a learning rate of $2e-5$ and split documents (title concatenated to the body) into passages of approximately 250 terms since fixed-size passages of 200–300 words are more effective than natural passages [31]. We use the TREC CAsT Tools[4] to split documents into passages. For inference, we process and concatenate all passages, and index the concatenated passages into Anserini indexes on which we retrieve with BM25.

[4]https://github.com/grill-lab/trec-cast-tools

**MonoBERT on Content.** Transformer-based re-rankers caused a paradigm shift in natural language processing and information retrieval [50]. Hence, we use MonoBERT [42], the first application of the well-known BERT [21] transformer to document-ranking, as a novel comparison baseline. We re-rank the top-100 documents retrieved by BM25 on the documents' content with the MonoBERT implementation of PyGaggle[5] using the default trained MonoBERT model castorini/monobert-large-msmarco, using the the maximum score of a passage as document score.

**MonoT5 on Content.** We use MonoT5 [43] to re-rank the top-100 documents retrieved by BM25 on the documents' content as a novel baseline on our navigational topics. MonoT5 classifies the relevance of a document to a given query using the sequence-to-sequence transformer T5 [45] and outperforms MonoBERT in experiments on MS MARCO and Robust04 [50]. We use the maximum score of a passage as the document score and use the implementation of MonoT5 provided in PyGaggle using the default trained MonoT5 model castorini/monot5-base-msmarco.

**BM25 on ORCAS.** We use the ORCAS query log [8] as a ranking baseline to continue our comparison of anchor text with query logs. We concatenate all queries clicked for the same target document and index these aggregated queries with Anserini [49]. The ORCAS query log is currently only available for version 1 of MS MARCO since it might cause train/test leakage for version 2 of MS MARCO in the 2021 TREC Deep Learning track.[6] Still, those concerns regarding train/test leakage do not apply to our situation, so we reuse the ORCAS query log on version 2 of MS MARCO.

**LambdaMART.** To study the effectiveness of anchor text in combination with other text types, we train four LambdaMART [4] models on different selections of features using the official training and validation labels for version 1 of the MS MARCO document collection. LambdaMART is the state-of-the-art feature-based learning to rank model [4, 28, 48], allowing us to analyze whether the observation that anchor text adds only small or no effectiveness in TREC style shared tasks [24] still holds. Overall, we calculate 32 features that come from four types of text: (1) our anchors, (2) ORCAS, (3) the title, and (4) the body. For each of the four text types, we calculate 8 features using Anserini [49] (TF, TF $\cdot$ IDF, BM25, F2exp, QL, QLJM, PL2, and SPL). We use LightGBM [32] to train four LambdaMART models on different selections of features: (1) using all 32 features ($\lambda$-MART@BTOA), (2) using body, title, and ORCAS ($\lambda$-MART@BTO), (3) using body, title, and the union of our anchor text ($\lambda$-MART@BTA), and (4) using only body and title ($\lambda$-MART@BT).

### 5.3 Evaluation

We experimentally compare the effectiveness of the 18 retrieval models on version 1 and version 2 of the document collection of MS MARCO. First, we reproduce the work of Craswell et al. [11] using our new 200 navigational topics, adding novel aspects to our reproduction with new baselines and evaluating the effectiveness of anchor text over time. Finally, we evaluate the 18 retrieval models on 43 + 45 informational topics

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

[6] https://microsoft.github.io/msmarco/TREC-Deep-Learning.html

**Table 4.** Overview of the mean reciprocal rank (MRR), recall at 3 (R@3), and recall at 10 (R@10) on 100 queries pointing to random entry pages and 100 queries pointing to popular entry pages on version 1 (V1) and version 2 of MS Marco (V2). The highest scores per group are bold.

| | | Random@V1 | | | Popular@V1 | | | Random@V2 | | | Popular@V2 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | MRR | R@3 | R@10 | MRR | R@3 | R@10 | MRR | R@3 | R@10 | MRR | R@3 | R@10 |
| **Anchor** | BM25@2016-07 | 0.61 | 0.63 | 0.68 | **0.62** | **0.72** | 0.83 | 0.56 | 0.61 | 0.64 | **0.57** | **0.64** | **0.80** |
| | BM25@2017-04 | 0.63 | 0.70 | 0.73 | 0.59 | 0.67 | 0.84 | 0.59 | 0.68 | 0.70 | 0.48 | 0.56 | 0.73 |
| | BM25@2018-13 | 0.70 | 0.76 | 0.82 | 0.54 | 0.65 | 0.81 | 0.62 | 0.68 | 0.77 | 0.47 | 0.54 | 0.77 |
| | BM25@2019-47 | 0.63 | 0.74 | 0.78 | 0.58 | 0.69 | 0.84 | 0.59 | 0.62 | 0.76 | 0.49 | 0.57 | 0.78 |
| | BM25@2020-05 | 0.63 | 0.72 | 0.79 | 0.55 | 0.66 | **0.86** | 0.56 | 0.64 | 0.71 | 0.45 | 0.53 | 0.74 |
| | BM25@2021-04 | 0.63 | 0.73 | 0.77 | 0.54 | 0.66 | 0.80 | 0.50 | 0.54 | 0.64 | 0.46 | 0.55 | 0.73 |
| | BM25@16–21 | **0.74** | **0.83** | **0.89** | 0.55 | 0.66 | 0.84 | **0.67** | **0.73** | **0.85** | 0.39 | 0.48 | 0.70 |
| **Content** | BM25@Content | 0.21 | 0.24 | 0.36 | 0.02 | 0.02 | 0.03 | 0.21 | 0.22 | 0.42 | 0.02 | 0.01 | 0.04 |
| | DeepCT@Anchor | **0.43** | **0.46** | **0.58** | **0.03** | **0.03** | 0.08 | **0.43** | **0.49** | **0.66** | 0.04 | 0.03 | **0.13** |
| | DeepCT@ORCAS | 0.38 | 0.42 | 0.57 | 0.02 | 0.00 | **0.09** | 0.36 | 0.40 | 0.60 | **0.05** | **0.04** | 0.10 |
| | DeepCT@Train | 0.27 | 0.28 | 0.44 | 0.02 | 0.01 | 0.05 | 0.32 | 0.34 | 0.49 | 0.03 | 0.02 | 0.08 |
| | MonoT5 | 0.39 | 0.43 | 0.53 | 0.02 | 0.01 | 0.05 | 0.38 | 0.43 | 0.57 | 0.04 | **0.04** | 0.08 |
| | MonoBERT | 0.35 | 0.37 | 0.51 | 0.02 | 0.01 | 0.05 | 0.36 | 0.41 | 0.56 | 0.01 | 0.01 | 0.02 |
| **Other** | BM25@ORCAS | **0.60** | **0.64** | **0.70** | **0.28** | **0.32** | **0.43** | **0.56** | 0.59 | 0.66 | **0.28** | **0.33** | **0.44** |
| | λ-MART@BTOA | 0.48 | 0.55 | 0.63 | 0.08 | 0.07 | 0.18 | 0.52 | 0.57 | **0.77** | 0.12 | 0.12 | 0.21 |
| | λ-MART@BTO | 0.41 | 0.49 | 0.57 | 0.07 | 0.06 | 0.17 | 0.49 | 0.55 | 0.65 | 0.08 | 0.10 | 0.14 |
| | λ-MART@BTA | 0.43 | 0.51 | 0.61 | 0.06 | 0.06 | 0.19 | 0.55 | **0.62** | 0.75 | 0.14 | 0.15 | 0.24 |
| | λ-MART@BT | 0.27 | 0.31 | 0.46 | 0.04 | 0.03 | 0.09 | 0.40 | 0.44 | 0.60 | 0.05 | 0.05 | 0.08 |

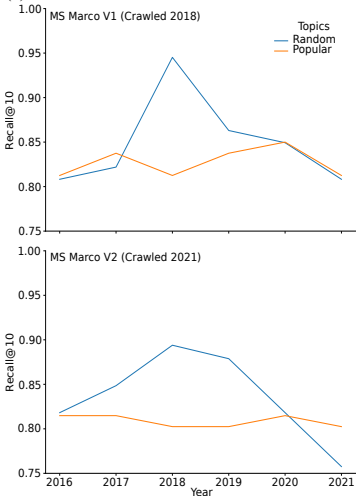
judged in the TREC Deep Learning track of 2019 and 2020. We report all significance tests using $p \leq 0.05$ and include the Bonferroni correction for multiple comparisons.

**Retrieval Effectiveness for Navigational Queries.** Table 4 shows the retrieval effectiveness in terms of the mean reciprocal rank (MRR), Recall@3, and Recall@10 for our 200 navigational topics on version 1 and version 2 of MS MARCO. Craswell et al. [11] report that anchor text achieves a statistically significant better MRR than the content. We can reproduce that anchor text achieves statistically significant improvements over methods using the documents' content, even for modern approaches.

For queries pointing to random entry pages, we observe that combining the anchor text of all years (BM25@16–21) achieves the best MRR of 0.74 on version 1 because this combination covers on average 0.56 million documents more than each single anchor text snapshot (see Table 1). While the MRR for single anchor text snapshots varies between 0.61 for 2016 and 0.70 for 2018, all anchor text approaches have a statistically significant higher MRR than DeepCT trained on anchor text, which achieves with an MRR of 0.43 the best performance among approaches retrieving solely on the document's content. Still, recent improvements in information retrieval are visible on our navigational topics, since DeepCT trained on anchor text, DeepCT trained on the ORCAS query log, MonoT5, MonoBERT, and three of the LambdaMART models improve statistically significant upon the MRR of 0.21 achieved by the BM25 retrieval on the content. BM25 on ORCAS improves statistically significantly upon all content-only models, even reaching the MRR scores of some anchor text samples. Still, the anchor text from 2018 and from 2016–2021 significantly improve upon ORCAS. Our results, which we observe similar on version 2 of MS MARCO, confirm the results by Craswell et al. [11] that anchor text is well suited for queries pointing to random entry-pages.

**Table 5.** (a) Overview of the effectiveness of anchor text on our navigational topics over the crawling period between 2016 and 2021. (b) Overview of the retrieval effectiveness on the TREC Deep Learning topics from 2019 and 2020 where we report nDCG@10 and nDCG@20.

(a) Effectiveness of Anchor Text over Time.




(b) Effectiveness on the TREC Deep Learning Topics.

| | | DL 2019 | | DL 2020 | |
|---|---|---|---|---|---|
| | | nDCG@10 | nDCG@20 | nDCG@10 | nDCG@20 |
| **Anchor** | BM25@2016-07 | 0.37 | 0.32 | 0.28 | 0.24 |
| | BM25@2017-04 | 0.39 | 0.35 | 0.29 | 0.27 |
| | BM25@2018-13 | 0.35 | 0.33 | 0.27 | 0.25 |
| | BM25@2019-47 | 0.34 | 0.31 | 0.27 | 0.25 |
| | BM25@2020-05 | 0.38 | 0.35 | 0.30 | 0.28 |
| | BM25@2021-04 | 0.35 | 0.32 | 0.30 | 0.26 |
| | BM25@16–21 | **0.41** | **0.38** | **0.34** | **0.32** |
| **Content** | BM25@Content | 0.51 | 0.50 | 0.53 | 0.53 |
| | DeepCT@Anchor | 0.53 | 0.54 | 0.55 | 0.53 |
| | DeepCT@ORCAS | 0.52 | 0.53 | 0.54 | 0.54 |
| | DeepCT@Train | 0.54 | 0.53 | 0.51 | 0.52 |
| | MonoT5 | **0.68** | **0.64** | **0.62** | **0.63** |
| | MonoBERT | 0.67 | 0.63 | **0.63** | **0.62** |
| **Other** | BM25@ORCAS | 0.45 | 0.41 | 0.36 | 0.33 |
| | λ-MART@BTOA | **0.59** | **0.55** | **0.57** | 0.57 |
| | λ-MART@BTO | 0.57 | **0.55** | **0.57** | **0.58** |
| | λ-MART@BTA | 0.57 | 0.54 | **0.57** | 0.57 |
| | λ-MART@BT | 0.57 | **0.55** | 0.56 | 0.56 |

For queries pointing to popular entry pages, all BM25 models retrieving on anchor text outperform all other retrieval models statistically significant. BM25 on the ORCAS query log follows with an MRR of 0.28 (both on version 1 and 2) which is significantly better than all non-anchor text methods, again highlighting some similarity of anchor text to query logs. Interestingly, we find that in almost all cases (the anchor text sample of 2016 being the only exception), queries pointing to popular entry pages are less effective than queries pointing to random entry pages. This observation contradicts the results by Craswell et al. [11] who reported an MRR of 0.45 (anchor) and 0.23 (content) for queries pointing to random entry pages, while queries pointing to popular entry pages achieve an higher MRR of 0.72 (anchor) and 0.37 (content). We manually inspected rankings and found the reason for this behavior in the fact that popular entry pages are the main subject of many articles with many occurrences of query terms.

**Retrieval Effectiveness of Anchor Text over Time.** To analyze the impact of the crawling time on the effectiveness of anchor text, we try to mitigate the effects of the crawler by removing all topics for which one of our anchor text samples retrieves less than 100 documents. This filtering step removes 47 of our 200 topics for version 1 (27 for random entry pages, 20 for popular entry pages) and 53 of our 200 topics for version 2 (34 for random entry pages, 19 for popular entry pages). The plot in Table 5a shows the effectiveness of anchor text between 2016 and 2021 for our remaining topics. The crawling time has only some negligible impact on the effectiveness for queries pointing to popular entry pages because popular pages obtain much anchor text and rarely change. On the other side, the crawling timestamp has a large impact on queries pointing to random entry pages. In particular, we observe a performance peak at 2018 with a Recall@10 in version 1 of 0.95 because version 1 of MS MARCO was crawled

in 2018. We also observe this peek in version 2 (crawled in 2021) because we transferred our topics from version 1 to version 2. Hence, our topics have a bias towards 2018 because we sampled the random entry pages in 2018. This bias exemplifies that the crawling time of anchor text has some relationship to the crawling time of the documents in the collection and to the query formulation time.

**Retrieval Effectiveness for Informational Queries.** We evaluate the effectiveness of our 18 retrieval models on informational queries using the TREC deep learning tracks of 2019 [14] and 2020 [13] on version 1 of MS MARCO (judgments for version 2 are not yet available). Since some of our retrieval models did not participate in the judgment pools, we removed all unjudged documents from the rankings to mitigate bias against those retrieval models as suggested by Sakai [46]. Table 5b shows the results in terms of nDCG@10 and nDCG@20. Unsurprisingly, modern transformers achieve the best scores, with MonoT5 reaching an nDCG@10 of 0.68 and MonoBERT reaching 0.67 in 2019. Our trained DeepCT models improve upon BM25, where DeepCT trained on anchor text performs similar to DeepCT trained on the ORCAS query log. All retrieval models solely using anchor text or the ORCAS query log—in contrast to their excellent effectiveness on navigational queries—are outperformed by BM25 on the content of the documents. Still, more anchor text is more effective, with the union of all anchor texts (BM25@16–21) achieving the best nDCG scores among all anchor text methods, almost reaching the effectiveness of ORCAS. While anchor text and the ORCAS query log are ineffective alone, our LambdaMART models show that they slightly improve nDCG@10 when compared to LambdaMART models that only use the body and title of documents ($\lambda$-MART@BTOA has an nDCG@10 of 0.59 while $\lambda$-MART@BT has 0.57 in 2019). Overall, our experiments confirm earlier observations [24] that anchor text is not effective in TREC style shared tasks covering exclusively informational queries.

## 6 Conclusion

In the scenario of the MS MARCO dataset, we have successfully reproduced the result of Craswell et al. [11] that anchor text is very effective for navigational queries. Trying to also reproduce the other seminal anchor text study of Eiron and McCurley [24] led to rather different results. We found that the term distributions of anchor texts and queries today are rather dissimilar and that retrieval against anchor text now yields less homogeneous results than retrieval against the document content.

Besides the above positive and negative reproducibility results, another important result of our study is that transformer-based approaches, be it in re-ranking scenarios or in the DeepCT context of estimating term importance, are less effective for navigational queries than a "basic" anchor text-oriented BM25 retrieval. Identifying navigational queries and for them switching to anchor text-based retrieval instead of neural models might thus improve the retrieval effectiveness of a general retrieval system. However, in the popular TREC Deep Learning tracks, the impact will be rather limited since the Deep Learning tracks do not involve navigational queries.

## Bibliography


[1] Bailey, P., Craswell, N., Hawking, D.: Engineering a Multi-Purpose Test Collection for Web Retrieval Experiments. Inf. Process. Manag. **39**(6), 853–871 (2003)

[2] Brin, S., Page, L.: The Anatomy of a Large-Scale Hypertextual Web Search Engine. Comput. Networks **30**(1-7), 107–117 (1998)

[3] Broder, A.Z.: A Taxonomy of Web Search. SIGIR Forum **36**(2), 3–10 (2002)

[4] Burges, C.J.: From RankNet to LambdaRank to LambdaMART: An Overview. Learning **11**(23-581), 81 (2010)

[5] Chen, W.F., Syed, S., Stein, B., Hagen, M., Potthast, M.: Abstractive Snippet Generation. In: Huang, Y., King, I., Liu, T., van Steen, M. (eds.) Proceedings of the World Wide Web Conference, WWW 2020, San Francisco, CA, USA, April 20-24, 2020, pp. 1309–1319, ACM (Apr 2020), ISBN 978-1-4503-7023-3

[6] Clarke, C.L.A., Craswell, N., Soboroff, I.: Overview of the TREC 2009 Web Track. In: Voorhees, E.M., Buckland, L.P. (eds.) Proceedings of the 18th Text REtrieval Conference, TREC 2009, Gaithersburg, MD, USA, November 17-20, 2009, NIST Special Publication, vol. 500-278, National Institute of Standards and Technology (NIST) (2009)

[7] Craswell, N., Billerbeck, B., Fetterly, D., Najork, M.: Robust Query Rewriting Using Anchor Data. In: Leonardi, S., Panconesi, A., Ferragina, P., Gionis, A. (eds.) Proceedings of the 6th ACM International Conference on Web Search and Data Mining, WSDM 2013, Rome, Italy, February 4-8, 2013, pp. 335–344, ACM (2013)

[8] Craswell, N., Campos, D., Mitra, B., Yilmaz, E., Billerbeck, B.: ORCAS: 20 Million Clicked Query-Document Pairs for Analyzing Search. In: d'Aquin, M., Dietze, S., Hauff, C., Curry, E., Cudré-Mauroux, P. (eds.) Proceedings of the 29th ACM International Conference on Information and Knowledge Management, CIKM 2020, Virtual Event, Ireland, October 19-23, 2020, pp. 2983–2989, ACM (2020)

[9] Craswell, N., Hawking, D.: Overview of the TREC-2002 Web Track. In: Voorhees, E.M., Buckland, L.P. (eds.) Proceedings of the 11th Text REtrieval Conference, TREC 2002, Gaithersburg, MD, USA, November 19-22, 2002, NIST Special Publication, vol. 500-251, National Institute of Standards and Technology (NIST) (2002)

[10] Craswell, N., Hawking, D.: Overview of the TREC 2004 Web Track. In: Voorhees, E.M., Buckland, L.P. (eds.) Proceedings of the 13th Text REtrieval Conference, TREC 2004, Gaithersburg, MD, USA, November 16-19, 2004, NIST Special Publication, vol. 500-261, National Institute of Standards and Technology (NIST) (2004)

[11] Craswell, N., Hawking, D., Robertson, S.E.: Effective Site Finding Using Link Anchor Information. In: Croft, W.B., Harper, D.J., Kraft, D.H., Zobel, J. (eds.) Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2001, New Orleans, LA, USA, September 9-13, 2001, pp. 250–257, ACM (2001)

[12] Craswell, N., Hawking, D., Wilkinson, R., Wu, M.: Overview of the TREC 2003 Web Track. In: Voorhees, E.M., Buckland, L.P. (eds.) Proceedings of the 12th Text REtrieval Conference, TREC 2003, Gaithersburg, MD, USA, November 18-21, 2003, NIST Special Publication, vol. 500-255, pp. 78–92, National Institute of Standards and Technology (NIST) (2003)

[13] Craswell, N., Mitra, B., Yilmaz, E., Campos, D.: Overview of the TREC 2020 Deep Learning Track. In: Voorhees, E.M., Ellis, A. (eds.) Proceedings of the 29th Text REtrieval Conference, TREC 2020, Virtual Event, Gaithersburg, MD, USA, November 16-20, 2020, NIST Special Publication, vol. 1266, National Institute of Standards and Technology (NIST) (2020)

[14] Craswell, N., Mitra, B., Yilmaz, E., Campos, D., Voorhees, E.M.: Overview of the TREC 2019 Deep Learning Track. CoRR **abs/2003.07820** (2020), URL https://arxiv.org/abs/2003.07820
[15] Craswell, N., Mitra, B., Yilmaz, E., Campos, D., Voorhees, E.M., Soboroff, I.: TREC Deep Learning Track: Reusable Test Collections in the Large Data Regime. In: Diaz, F., Shah, C., Suel, T., Castells, P., Jones, R., Sakai, T. (eds.) Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2021, Virtual Event, Canada, July 11-15, 2021, pp. 2369–2375, ACM (2021)
[16] Croft, W.B., Metzler, D., Strohman, T.: Search Engines - Information Retrieval in Practice. Pearson Education (2009), ISBN 978-0-13-136489-9
[17] Dai, N., Davison, B.D.: Mining Anchor Text Trends for Retrieval. In: Gurrin, C., He, Y., Kazai, G., Kruschwitz, U., Little, S., Roelleke, T., Rüger, S.M., van Rijsbergen, K. (eds.) Proceedings of the 32nd European Conference on IR Research, ECIR 2010, Milton Keynes, UK, March 28-31, 2010, Lecture Notes in Computer Science, vol. 5993, pp. 127–139, Springer (2010)
[18] Dai, Z., Callan, J.: Context-Aware Sentence/Passage Term Importance Estimation For First Stage Retrieval. CoRR **abs/1910.10687** (2019)
[19] Dai, Z., Callan, J.: Context-Aware Document Term Weighting for Ad-Hoc Search. In: Huang, Y., King, I., Liu, T., van Steen, M. (eds.) Proceedings of the World Wide Web Conference, WWW 2020, Taipei, Taiwan, April 20-24, 2020, pp. 1897–1907, ACM / IW3C2 (2020)
[20] Dang, V., Croft, W.B.: Query Reformulation Using Anchor Text. In: Davison, B.D., Suel, T., Craswell, N., Liu, B. (eds.) Proceedings of the 3rd ACM International Conference on Web Search and Web Data Mining, WSDM 2010, New York, NY, USA, February 4-6, 2010, pp. 41–50, ACM (2010)
[21] Devlin, J., Chang, M., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Burstein, J., Doran, C., Solorio, T. (eds.) Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1, pp. 4171–4186, Association for Computational Linguistics (2019)
[22] Dou, Z., Song, R., Nie, J., Wen, J.: Using Anchor Texts with their Hyperlink Structure for Web Search. In: Allan, J., Aslam, J.A., Sanderson, M., Zhai, C., Zobel, J. (eds.) Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009, Boston, MA, USA, July 19-23, 2009, pp. 227–234, ACM (2009)
[23] Dunlop, M.D., van Rijsbergen, C.J.: Hypermedia and Free Text Retrieval. Inf. Process. Manag. **29**(3), 287–298 (1993)
[24] Eiron, N., McCurley, K.S.: Analysis of Anchor Text for Web Search. In: Clarke, C.L.A., Cormack, G.V., Callan, J., Hawking, D., Smeaton, A.F. (eds.) Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2003, Toronto, Canada, July 28-August 1, 2003, pp. 459–460, ACM (2003)
[25] Fuglede, B., Topsøe, F.: Jensen-Shannon Divergence and Hilbert Space Embedding. In: Proceedings of the 2004 IEEE International Symposium on Information Theory, ISIT 2004, Chicago Downtown Marriott, Chicago, IL, USA, June 27-July 2, 2004, p. 31, IEEE (2004)
[26] Hawking, D.: Overview of the TREC-9 Web Track. In: Voorhees, E.M., Harman, D.K. (eds.) Proceedings of the 9th Text REtrieval Conference, TREC 2000, Gaithersburg, MD, USA, November 13-16, 2000, NIST Special Publication, vol. 500-249, National Institute of Standards and Technology (NIST) (2000)

[27] Hawking, D., Voorhees, E.M., Craswell, N., Bailey, P.: Overview of the TREC-8 Web Track. In: Voorhees, E.M., Harman, D.K. (eds.) Proceedings of the 8th Text REtrieval Conference, TREC 1999, Gaithersburg, MD, USA, November 17-19, 1999, NIST Special Publication, vol. 500-246, National Institute of Standards and Technology (NIST) (1999)
[28] Hu, Z., Wang, Y., Peng, Q., Li, H.: Unbiased LambdaMART: An Unbiased Pairwise Learning-to-Rank Algorithm. In: Liu, L., White, R.W., Mantrach, A., Silvestri, F., McAuley, J.J., Baeza-Yates, R., Zia, L. (eds.) Proceedings of the World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019, pp. 2830–2836, ACM (2019)
[29] Kamps, J., Kaptein, R., Koolen, M.: Using Anchor Text, Spam Filtering and Wikipedia for Web Search and Entity Ranking. In: Voorhees, E.M., Buckland, L.P. (eds.) Proceedings of the 19th Text REtrieval Conference, TREC 2010, Gaithersburg, MD, USA, November 16-19, 2010, NIST Special Publication, vol. 500-294, National Institute of Standards and Technology (NIST) (2010)
[30] Karpukhin, V., Oguz, B., Min, S., Lewis, P.S.H., Wu, L., Edunov, S., Chen, D., Yih, W.: Dense Passage Retrieval for Open-Domain Question Answering. In: Webber, B., Cohn, T., He, Y., Liu, Y. (eds.) Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Virtual Event, November 16-20, 2020, pp. 6769–6781, Association for Computational Linguistics (2020)
[31] Kaszkiel, M., Zobel, J.: Passage Retrieval Revisited. In: Belkin, N.J., Narasimhalu, A.D., Willett, P., Hersh, W.R., Can, F., Voorhees, E.M. (eds.) Proceedings of the 20th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 1997, Philadelphia, PA, USA, July 27-31, 1997, pp. 178–185, ACM (1997)
[32] Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., Liu, T.: LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In: Guyon, I., von Luxburg, U., Bengio, S., Wallach, H.M., Fergus, R., Vishwanathan, S.V.N., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA, pp. 3146–3154 (2017)
[33] Kilgarriff, A., Rose, T.: Measures for Corpus Similarity and Homogeneity. In: Ide, N., Voutilainen, A. (eds.) Proceedings of the 3rd Conference on Empirical Methods for Natural Language Processing, Palacio de Exposiciones y Congresos, Granada, Spain, June 2, 1998, pp. 46–52, ACL (1998)
[34] Kobayashi, M., Takeda, K.: Information Retrieval on the Web. ACM Comput. Surv. **32**(2), 144–173 (2000)
[35] Koolen, M., Kamps, J.: The Importance of Anchor Text for Ad Hoc Search Revisited. In: Crestani, F., Marchand-Maillet, S., Chen, H., Efthimiadis, E.N., Savoy, J. (eds.) Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2010, Geneva, Switzerland, July 19-23, 2010, pp. 122–129, ACM (2010)
[36] Kraft, R., Zien, J.Y.: Mining Anchor Text for Query Refinement. In: Feldman, S.I., Uretsky, M., Najork, M., Wills, C.E. (eds.) Proceedings of the 13th International World Wide Web Conference, WWW 2004, New York, USA, May 17-20, 2004, pp. 666–674, ACM (2004)
[37] Lin, J., Yang, P.: The Impact of Score Ties on Repeatability in Document Ranking. In: Piwowarski, B., Chevalier, M., Gaussier, É., Maarek, Y., Nie, J., Scholer, F. (eds.) Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019, pp. 1125–1128, ACM (2019)

[38] Ma, Z., Dou, Z., Xu, W., Zhang, X., Jiang, H., Cao, Z., Wen, J.: Pre-training for Ad-hoc Retrieval: Hyperlink is Also You Need. In: 30th ACM International Conference on Information and Knowledge Management (CIKM 2021), ACM (Nov 2021)
[39] Macdonald, C., Tonellotto, N.: Declarative Experimentation in Information Retrieval using PyTerrier. In: Balog, K., Setty, V., Lioma, C., Liu, Y., Zhang, M., Berberich, K. (eds.) ICTIR '20: The 2020 ACM SIGIR International Conference on the Theory of Information Retrieval, Virtual Event, Norway, September 14-17, 2020, pp. 161–168, ACM (2020)
[40] McBryan, O.A.: GENVL and WWWW: Tools for Taming the Web. In: Proceedings of the 1st International World Wide Web Conference, WWW 1994, Geneva, Switzerland, May 25-27, 1994, vol. 341 (1994)
[41] Metzler, D., Novak, J., Cui, H., Reddy, S.: Building Enriched Document Representations Using Aggregated Anchor Text. In: Allan, J., Aslam, J.A., Sanderson, M., Zhai, C., Zobel, J. (eds.) Proceedings of the 32nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2009, Boston, MA, USA, July 19-23, 2009, pp. 219–226, ACM (2009)
[42] Nogueira, R., Cho, K.: Passage Re-ranking with BERT. CoRR **abs/1901.04085** (2019)
[43] Nogueira, R., Jiang, Z., Pradeep, R., Lin, J.: Document Ranking with a Pretrained Sequence-to-Sequence Model. In: Cohn, T., He, Y., Liu, Y. (eds.) Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Virtual Event, November 16-20, 2020, Findings of ACL, vol. EMNLP 2020, pp. 708–718, Association for Computational Linguistics (2020)
[44] Ogilvie, P., Callan, J.P.: Combining Document Representations for Known-Item Search. In: Clarke, C.L.A., Cormack, G.V., Callan, J., Hawking, D., Smeaton, A.F. (eds.) Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2003, July 28-August 1, 2003, Toronto, ON, Canada, pp. 143–150, ACM (2003)
[45] Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. J. Mach. Learn. Res. **21**, 140:1–140:67 (2020)
[46] Sakai, T.: Alternatives to Bpref. In: Kraaij, W., de Vries, A.P., Clarke, C.L.A., Fuhr, N., Kando, N. (eds.) Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2007, Amsterdam, The Netherlands, July 23-27, 2007, pp. 71–78, ACM (2007)
[47] Westerveld, T., Kraaij, W., Hiemstra, D.: Retrieving Web Pages Using Content, Links, URLs and Anchors. In: Voorhees, E.M., Harman, D.K. (eds.) Proceedings of the 10th Text REtrieval Conference, TREC 2001, Gaithersburg, MD, USA, November 13-16, 2001, NIST Special Publication, vol. 500-250, National Institute of Standards and Technology (NIST) (2001)
[48] Wu, Q., Burges, C.J.C., Svore, K.M., Gao, J.: Adapting Boosting for Information Retrieval Measures. Inf. Retr. **13**(3), 254–270 (2010)
[49] Yang, P., Fang, H., Lin, J.: Anserini: Enabling the Use of Lucene for Information Retrieval Research. In: Kando, N., Sakai, T., Joho, H., Li, H., de Vries, A.P., White, R.W. (eds.) Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2017, Shinjuku, Tokyo, Japan, August 7-11, 2017, pp. 1253–1256, ACM (2017)
[50] Yates, A., Nogueira, R., Lin, J.: Pretrained Transformers for Text Ranking: BERT and Beyond. In: Diaz, F., Shah, C., Suel, T., Castells, P., Jones, R., Sakai, T. (eds.) Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2021, Virtual Event, Canada, July 11-15, 2021, pp. 2666–2668, ACM (2021)