CopyCat: Near-Duplicates within and between the ClueWeb and the Common Crawl

Maik Fröbe* Janek Bevendorff† Lukas Gienapp‡ Michael Völske†
Benno Stein† Martin Potthast‡ Matthias Hagen*

*Martin-Luther-Universität Halle-Wittenberg, †Bauhaus-Universität Weimar, ‡Leipzig University

ABSTRACT

The amount of near-duplicates in web crawls like the ClueWeb or Common Crawl demands from their users either to develop a preprocessing pipeline for deduplication, which is costly both computationally and in person hours, or accepting the undesired effects that near-duplicates have on reliability and validity of experiments. We introduce ChatNoir-CopyCat-21, which simplifies deduplication significantly. It comes in two parts: (1) A compilation of near-duplicate documents *within* the ClueWeb09, the ClueWeb12, and two Common Crawl snapshots, as well as *between* selections of these crawls, and (2) a software library that implements the deduplication of arbitrary document sets. Our analysis shows that 14–52 % of the documents within a crawl and around 0.7–2.5 % between the crawls are near-duplicates. Two showcases demonstrate the application and usefulness of our resource.

CCS CONCEPTS

• **Information systems → Evaluation of retrieval results**.

KEYWORDS

Near-duplicate detection; TREC evaluation; Relevance Transfer

ACM Reference Format:

Maik Fröbe, Janek Bevendorff, Lukas Gienapp, Michael Völske, Benno Stein, Martin Potthast, and Matthias Hagen. 2021. CopyCat: Near-Duplicates within and between the ClueWeb and the Common Crawl. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '21), July 11–15, 2021, Virtual Event, Canada.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3404835.3463246

1 INTRODUCTION

Web crawls typically contain a high number of pages that are duplicates or near-duplicates [11], i.e., documents with different URLs and identical or very similar content. Google and other web search engines address this issue by identifying them at crawl time [20], returning only the presumed original version from a set of near-duplicates for a query.[1] For web crawls commonly used in academia—most notably the ClueWeb09, the ClueWeb12,[2] and the Common Crawls[3]—no systematic duplication analysis has been carried out so far. In fact, by unwritten consensus barring few exceptions, the costs of applying deduplication in information retrieval experiments are often traded for the incalculable but real threats to an experiment's reliability and validity. Bernstein and Zobel [5] raised red flags on this issue long ago; recently reproduced [14], it has been shown that including near-duplicates strongly affects a retrieval system's measured effectiveness. For learning to rank approaches, including near-duplicates is akin to oversampling, biasing the trained models [13]. Retrieval experiments in academia are typically carried out by few individuals on a comparably small computing infrastructure, and there is a lack of reference deduplication libraries as well as of best-practice examples to follow. Though this may explain the current modus operandi, we should strive for improvement nonetheless.

Table 1: CopyCat at a glance: Near-duplicates found in the ClueWeb09 (cw09), the ClueWeb12 (cw12), and the Common Crawls 2015-11 (cw15) and 2017-04 (cw17).

Statistic	Web crawl				Σ
	cw09	cw12	cc15	cc17	
Compressed size	4.0 TB	4.5 TB	28.1 TB	54.0 TB	90.6 TB
Documents	1.0 b	731.7 m	1.8 b	3.1 b	6.7 b
SimHash duplicates	145.6 m	201.2 m	907.2 m	1.0 b	2.3 b
Canonical links	3.0 m	67.1 m	747.5 m	1.5 b	2.3 m
Crawled duplicates	1.5 m	11.2 m	278.3 m	180.1 m	471.1 m
CopyCat duplicates	145.8 m	204.3 m	951.2 m	1.0 b	2.3 b

As a step into this direction we contribute ChatNoir-CopyCat-21, or CopyCat for short, as a resource to address the aforementioned issues. By analyzing canonical links and systematically applying the state-of-the-art SimHash approach to the mentioned web crawls (cf. Sections 3 and 4), we compile lists of near-duplicate documents for efficient deduplication. As showcases, we analyze TREC runs and qrels (cf. Section 5), and put, for the first time, the transfer of relevance judgments from older to newer crawls into practice to enrich the ClueWeb12 with 162 and the Common Crawl 2015-11 with 138 sparsely judged topics (cf. Section 5). Both data and software are provided under an open source license.[4,5] CopyCat enables easy deduplication of web crawls, either by filtering pages using our list of pre-computed near-duplicates, or by running the deduplication pipeline on a fresh crawl. We also paid special attention to efficiently support deduplication in TREC-style evaluations.

Table 1 summarizes the analyzed web crawls[6] and the near-duplicates found within. The adoption of canonical links to indicate

[1] https://developers.google.com/search/docs/advanced/guidelines/duplicate-content
[2] https://lemurproject.org/clueweb09.php/ and https://lemurproject.org/clueweb12/

SIGIR '21, July 11–15, 2021, Virtual Event, Canada

ACM ISBN 978-1-4503-8037-9/21/07…$15.00
https://doi.org/10.1145/3404835.3463246

[3] https://commoncrawl.org/
[4] http://webis.de/data/chatnoir-copycat-21
[5] http://github.com/chatnoir-eu/chatnoir-copycat-21
[6] The selection of crawls follows that of the ChatNoir research search engine [6].

duplicated pages has increased over time from 0.3 % (cw09) to 48 % (cc17). When exclusively using canonical links for near-duplicate detection, our analysis shows that the recall is insufficient for practical use. Our improved version of the SimHash algorithm identifies substantially more near-duplicates at higher precision. Altogether, CopyCat compiles a list of 2.3 billion near-duplicates from the four crawls, which amounts to about a third (34 %) of all crawled pages.

2 RELATED WORK

This section reviews the definitions for near-duplicates, the assessments of their prevalence on the web and the issues caused, and the existing approaches to near-duplicate detection at scale.

Defining Near-Duplicates. The fact that there is no universal definition of what a near-duplicate is renders their detection and a comparable analysis difficult. Henzinger [15] and Manku et al. [20] apply the most restrictive definition of near-duplicates to tune and evaluate their algorithms. They consider document pairs as near-duplicates if they differ only by their session or message IDs, timestamps, visitor counts, server names, invisible differences, or URL parts, or if they are entry pages to the same site. Note that the crawls that we consider were collected over months, but that documents with minor content changes (which are likely to appear at different crawling dates) are often not considered near-duplicates under Henzinger's definition. Bernstein and Zobel [5] give a more relaxed definition, allowing for content changes if the respective documents are only "different versions of the same article," i.e., if a user will get the same information from both documents for all "reasonable queries." We adopt this definition since it is rooted in the very content: a pair of pages form duplicates if they are equivalent in terms of the information needs to which they are relevant.

Prevalence of Near-Duplicates on the Web. A large portion of web content is duplicated [11]. Though a substantial number of web pages change regularly, the consecutive versions are usually highly similar [9]. Fetterly et al. [11, 12] investigated the similarities between updated documents over eleven weeks and found that most of the changes to the content were negligible, with 30 % of web pages being near-duplicates. Ntoulas et al. [21] tracked the link structures of 150 domains over one year, also noticing only insignificant changes on most pages. Adar et al. [1] repeatedly crawled 55 000 URLs over 5 weeks and analyzed the similarities between the pages: Two-thirds of the pages had changed content, but most of these changes were minimal, confirming earlier studies [22].

The URLs under which web pages are accessible are mostly stable: Ntoulas et al. [21] observe that half of their 150 crawled pages were still accessible after six weeks, and 40 % after one year. This effect was even more pronounced in studies spanning larger numbers of web pages: Fetterly et al. observe that 88 % of their 150 million pages were still accessible after eleven weeks [11, 12]. Similar results were observed for the setting of Kim and Lee [16]: Once a page was downloaded successfully, its URL continued to be retrievable in subsequent attempts in most cases. As much as 73 % of their crawl (covering 3 million URLs) remained accessible during a 100 day crawling period.

The crawling timestamps of the crawls that we consider span a time period that has not been covered by the literature. Nevertheless, the amount of near-duplicates and the URL stability reported in existing studies serve as motivation in two regards. First, to identify and remove near-duplicates from web crawls used in academia, and, secondly, to trace documents with relevance judgments in a TREC track, since they might still occur as near-duplicates in later crawls.

Issues Caused by Near-Duplicates. Oversampling data before partitioning it into training and test sets can invalidate evaluations because a model sees the same object during training and test [25]. Near-duplicates may cause such an information leakage easily. I.e., a model may overfit on groups of near-duplicates in the training set [13] or find pages in the test set that duplicate a training document, distorting a researcher's perception of generalizability.

Bernstein and Zobel [5] argue that near-duplicates cause additional problems in information retrieval evaluations because search engine users do not benefit from seeing near-duplicates. To this end, they introduce the novelty principle to mark a document as irrelevant when a near-duplicate is ranked higher. The show that applying the novelty principle decreases mean average precision scores by 20 % on average. Actually the problem persists to this day since web crawls contain many near-duplicates that strongly influence the rankings of retrieval systems [14].

Near-Duplicate Detection. There are three types of algorithms for detecting near-duplicates [3]: (1) syntactic, (2) URL-based, and (3) semantic ones. While URL-based deduplication is attractive for deduplicating large crawls at moderate costs [2, 18], we focus on syntactic near-duplicates, since they allow for a precision-oriented baseline. There are many effective solutions to detecting syntactic near-duplicates based on fingerprinting techniques [7, 8, 15, 20]. Manku et al. [20] demonstrated that the SimHash algorithm is able to handle near-duplicate detection even with a small fingerprint size of 64 bit and a Hamming-threshold of 3 bit, while maintaining a good precision/recall tradeoff [20].

The approach of Henzinger [15] allows to reduce the number of pairwise similarities that must be calculated with SimHash. Henzinger exploits that the Hamming distances of the hashes of near-duplicates are always less or equal to k. Consequently, partitioning the hashes into $k + 1$ disjoint subsets ensures that one subset is always identical for near-duplicates. This reduces the number of pairwise comparisons, as only hashes with at least one identical subset must be considered. Henzinger demonstrated the efficiency of this approach on a proprietary billion-page crawl.

Following the literature, we use a 64-bit SimHash with a 3-bit near-duplicate threshold and employ Henzinger's partitioning trick to reduce the number of pairwise comparisons.

3 CANONICAL LINKS

Canonical links—introduced in 2012—allow authors of web pages to indicate duplicate content.[7] A canonical link's target page "must identify content that is either duplicative or a superset of the content" of the source page, indicating that the target page is the preferred version. Search engines may use this information to resolve duplicate search results. This raises the question of whether canonical links already solve the issues caused by near-duplicates.

Analysis. As discussed at the outset, Table 1 overviews the total of canonical links found in the web crawls, showing that their adoption has surged. Table 2a lists the top domains with the highest

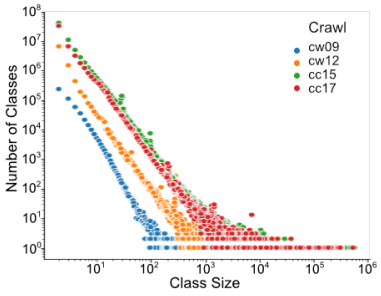
[7] RFC 6596 https://tools.ietf.org/html/rfc6596

Table 2: (a) Overview of domains with the most canonical links. (b) Distribution of equivalence classes of canonical links. (c) Syntactic similarity of document pairs from the same equivalence class. (d) Overview of pairs sampled for the pilot study.

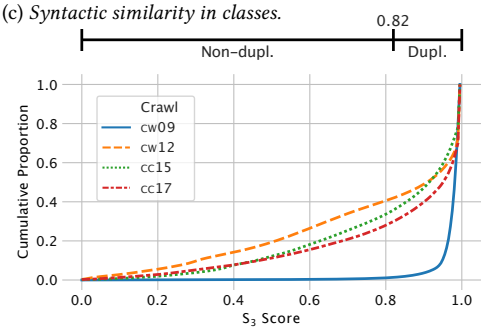
(a) *Overview of domains (opendns.com tag).*

	Domain	Docs	Classes
cw09	en.wikipedia.org *(Research)*	2.0 m	0.5 m
	automobilsport.com *(Sports)*	2.5 k	1
	campingcompass.com *(Travel)*	1.9 k	1
cw12	beyond.com *(Adware)*	0.4 m	3.5 k
	skiset.co.uk *(Travel)*	0.3 m	81
	php.net *(Tech)*	0.2 m	16.0 k
cc15	urbandictionary.com *(Humor)*	9.2 m	0.3 m
	m.mlb.com *(Sports)*	1.9 m	0.2 m
	agoda.com *(Travel)*	1.0 m	0.2 m
cc17	urbandictionary.com *(Humor)*	2.8 m	0.2 m
	merkur.de *(News)*	1.7 m	0.2 m
	tz.de *(News)*	1.1 m	0.2 m

(b) *Distribution of classes.*



(c) *Syntactic similarity in classes.*



(d) *Overview of pairs sampled for the pilot study.*

	cw09	cw12	cc15	cc17
Documents (Classes)	3.0 m (1.4 m)	99.9 m (79.3 m)	29.1 m (7.0 m)	19.4 m (5.1 m)
Pairs	3.7 m	85.3 m	172.2 m	99.8 m

number of documents that contain canonical links per crawl. Some of the high-ranked domains host a staggering 75–99 % of redundant documents. Apparently, different crawling strategies tend to collect different highly-redundant domains. As a result, the top domains differ among the ClueWeb crawls (albeit, when they were crawled, canonical links were not standardized, yet), whereas both Common Crawls have urbandictionary.com as their most redundant domain.

Documents with canonical links pointing to the same target document form an equivalence class. Table 2b provides an overview of the sizes of the equivalence classes formed by canonical links in the four web crawls. The equivalence class sizes over the number of equivalence classes follow a power law distribution. The same distribution has been reported for equivalence classes of near-duplicates [11, 18]. Both Common Crawls have equivalence classes comprising more than 100 000 documents.

Verification. As a pilot study to verify whether the canonical link document pairs are indeed near-duplicates, we employ the lossless S_3 fingerprint similarity of Bernstein and Zobel [4] using word-8-grams. An S_3 score of 0 indicates no overlap between documents, an S_3 score of 1 means equality. The CopyCat software library implements this algorithm and supports all preprocessing steps provided by the Anserini information retrieval toolkit [27], as well as main content extraction implemented by ChatNoir [6] and Boilerpipe [17] to convert raw web pages to text. To conform with the preprocessing steps of previous large-scale near-duplicate studies that used SimHash [15, 20], we use the default plaintext from HTML extraction of Anserini. A subsequent process removes stop words using Lucene's default stop word list for English, applies stemming with the Porter Stemmer, and lowercases the remaining words. These preprocessing steps are widely used in practice since they correspond to Anserini's default for the ClueWebs.

We calculate the S_3 scores for all document pairs among 50 randomly selected documents per equivalence class of the ClueWeb crawls and a random 10 % sample of the Common Crawl classes. Table 2d provides an overview of our sample and Table 2c shows the cumulative proportion of pairs from the sampled equivalence classes above a given S_3 score for each of the crawls. Documents from the same canonical link class do indeed often have high syntactic similarity in terms of their S_3 scores. For the ClueWeb09, 96 % of the canonical link pairs have an S_3 score above 0.9—mainly due to the well-placed canonical links in Wikipedia, one of the early adopters. However, the later crawls contain some substantial amount of pairs with lower S_3 scores. This indicates that canonical links can be a suitable indicator for high syntactic similarity but actual checks using the content are still required.

To identify an appropriate S_3 threshold that corresponds to near-duplicates and conforms to previous research [13, 14], we sample 100 document pairs belonging to the same equivalence class for each crawl, which uniformly cover S_3 scores between 0.4 and 1 for manual review. We use the near-duplicate definition and review guidelines of Bernstein and Zobel [5] to label near-duplicates: A document pair is considered as near-duplicates when both documents are content-equivalent, and users would be able to extract the same information from either one for all reasonable queries. Two versions of the same Wikipedia article with only minor non-content changes are an example of near-duplicates under this definition. We find that pairs with $S_3 \geq 0.82$ are near-duplicates in our reviewed sample with a precision of 0.95. Overall, 65 % of document pairs from the same canonical link class are near-duplicates using an S_3 threshold of 0.82 (min: 58 % in cw12; max: 98 % in cw09).

Discussion. Our manual assessment reveals that canonical links alone are insufficient for labeling near-duplicates. We found numerous false positives, e.g., landing pages with changed featured content, or different versions of a page crawled at different points in time. Additionally, we also identified cases in which—even for sites that otherwise make correct use of canonical links—near-duplicates are not marked with canonical links. Thus, the pages from these pairs end up in different link classes, when in fact their equivalence classes should be merged. Given these inaccuracies, canonical links can serve as a valuable low-effort baseline for identifying near-duplicates, but are insufficient on their own. We therefore augment our near-duplicate detection with a SimHash-based method.

Table 3: (a) Overview of the precision, recall, and F_1 score of different document vector representations for SimHash within classes of canonical links. (b) Overview of the precision of document vector representations for SimHash within the crawls. For both tables, the maximum per column segment is marked in bold, and rows of used features are marked in italics.

(a) *Effectiveness in classes of canonical links.*

Feature	cw09			cw12			cc15			cc17		
	Prec.	Rec.	F_1	Prec.	Rec.	F_1	Prec.	Rec.	F_1	Prec.	Rec.	F_1
1-grams	***1.00***	***0.98***	***0.99***	*0.95*	***0.89***	***0.92***	*0.94*	***0.82***	***0.88***	*0.96*	***0.88***	***0.92***
3-grams	**1.00**	0.78	0.87	**0.99**	0.71	0.83	0.99	0.60	0.75	0.99	0.69	0.82
5-grams	**1.00**	0.67	0.80	1.00	0.65	0.79	**1.00**	0.53	0.69	**1.00**	0.63	0.77
8-grams	**1.00**	0.56	0.72	1.00	0.60	0.75	**1.00**	0.47	0.64	**1.00**	0.57	0.73
1–3-grams	**1.00**	**0.94**	**0.97**	0.98	**0.83**	**0.90**	0.97	**0.76**	**0.85**	0.98	**0.82**	**0.89**
1–5-grams	**1.00**	0.91	0.95	0.98	0.81	0.89	0.97	0.73	0.83	0.98	0.80	0.88
1–8-grams	**1.00**	0.89	0.94	0.99	0.79	0.88	0.97	0.71	0.82	0.98	0.78	0.87
3–5-grams	***1.00***	*0.73*	*0.84*	***1.00***	*0.68*	*0.81*	***1.00***	*0.57*	*0.72*	*0.99*	*0.66*	*0.79*
3–8-grams	**1.00**	0.67	0.80	**1.00**	0.65	0.79	**1.00**	0.53	0.70	0.99	0.63	0.77
5–8-grams	**1.00**	0.62	0.76	**1.00**	0.62	0.77	**1.00**	0.50	0.67	**1.00**	0.60	0.75

(b) *Precision in crawls.*

Feature	Precision			
	cw09	cw12	cc15	cc17
1-grams	*0.60*	*0.35*	*0.65*	*0.58*
3-grams	0.98	0.90	0.94	0.88
5-grams	**1.00**	0.92	**0.99**	0.89
8-grams	**1.00**	**0.95**	**0.99**	**0.95**
1-3-grams	0.82	0.45	0.75	0.66
1–5-grams	0.69	0.44	0.73	0.66
1–8-grams	0.81	0.44	0.72	0.63
3–5-grams	*0.99*	*0.97*	***0.99***	***0.97***
3–8-grams	0.99	0.97	0.98	0.96
5–8-grams	**1.00**	**0.98**	**0.99**	0.88

4 DEDUPLICATION OF COMPLETE CRAWLS

For a given document collection, CopyCat identifies near-duplicates in four steps: (1) calculation of the SimHash fingerprint for each document, (2) selection of one representative document, if multiple documents have identical fingerprints, (3) partitioning the remaining fingerprints, (4) calculation of the Hamming distances between all fingerprints of a partition. Documents with identical fingerprints (Hamming distance of 0 bits) become part of the CopyCat dataset. Selecting only one representative per group of documents with identical fingerprints reduces the number of pairwise comparisons.

We follow known best practices and configure our SimHash implementation to output 64-bit fingerprints at a near-duplicate threshold of 3 bits [20]. Documents are preprocessed the same way as in the pilot study by extracting their plain text, removing stop words, stemming, and lowercasing the remaining words. To choose a suitable vector representation of the processed documents, we experiment with word-1-grams, word-3-grams, word-5-grams, word-8-grams, as well as any pairwise combinations of these.

Near-Duplicate Detection vs. Canonical Links. Table 3a shows the precision, recall, and F_1 scores of our approach for all document vector representations when using equivalence classes of canonical links as input. Within an equivalence class (see Table 2c), a pair of documents is considered as near-duplicates if its S_3 score is above the empirically determined threshold of 0.82. Due to the reduced number of document pairs in this setting, we can calculate the recall of all document vector representations. The considered document vector representations achieve a remarkably high precision. Even in the worst case (Common Crawl 2015-11 with word-1-gram features), a precision of 0.94 is measured. Overall, word-1-grams consistently obtain the best recall by a margin compared to other feature sets and lead to the best F_1 scores between 0.88 and 0.99. Hence we use word-1-grams as the features for near-duplicate detection within canonical link equivalence classes.

Scaling Near-Duplicate Detection. Less than half of all documents from the four crawls come with canonical links. To find suitable parameters for scaling the classification to the remaining documents, we reused documents from existing equivalence classes (Table 2d), but this time removed the constraint that only documents from the same class are considered near-duplicates. For each potential document vector representation, all candidate pairs with Hamming distances ≤ 3 were identified, and from these 50 000 pairs were sampled at random. Finally, we considered any document pair as near-duplicates whose S_3 score exceeded the threshold of 0.82. Recall cannot be calculated in this setting, since identifying the S_3 scores for all document pairs is computationally infeasible. Table 3b shows the precision obtained. Word-1-grams achieved the lowest precision, where the ClueWeb12 constituted the worst case overall with a precision of 0.35. As best-performing representations, we proceeded with a combination of word-3- and -5-grams for the task of deduplicating the entire crawls. Their precision is comparable to that of word-8-grams (Table 3b), while yielding a better recall when canonical links are available (Table 3a).

CopyCat Construction. The final CopyCat dataset combines both deduplication approaches in the form of inclusion and exclusion lists of near-duplicates, which researchers can use for duplicate-free versions of the ClueWeb crawls or the two Common Crawl versions. Documents on the inclusion list are not near-duplicates of each other, whereas documents on the exclusion list have a near-duplicate on the inclusion list. The representative document of a class is always the one with the alphanumerically lowest ID; all others are considered its near-duplicates. This is a common tie breaking mechanism in IR [27], and with randomly chosen IDs, leads to a good random distribution regarding other document properties such as the text length [19].

We ran CopyCat on each of the four crawls (14–52 % near-duplicates; see Table 1), the category B subsets of the ClueWeb crawls (12 % near-duplicates), and the union of the ClueWeb09, the ClueWeb12, and the Common Crawl 2015-11 to analyze duplication between the crawls. Up to 26.4 m documents from the ClueWeb09 have near-duplicates in the ClueWeb12, which corresponds to 2.5 % of the ClueWeb09. Furthermore, there are 6.9 m ClueWeb09 documents (0.7 %) and 12.4 m ClueWeb12 documents (1.7 %) that have near-duplicates in the Common Crawl 2015-11. The existence of near-duplicates between these crawls motivates our investigation of whether they include judged documents from TREC tracks.

Table 4: Overview of the proportion of near-duplicates at an S_3 threshold of 0.82 in the relevance judgments and in the top-k results of submitted runs for the TREC web tracks.

Web track			Near-dupl. in judgment			Near-dupl. in runs		
Year	Runs	Judg.	All	Relevant	Irrelevant	@10	@100	@1000
2009	71	13 118	0.14	0.18	0.11	0.11	0.17	0.19
2010	56	25 329	0.17	0.21	0.15	0.19	0.25	0.25
2011	37	19 381	0.19	0.21	0.19	0.21	0.24	0.25
2012	28	16 055	0.16	0.25	0.13	0.20	0.18	0.20
2013	34	14 474	0.17	0.13	0.17	0.12	0.19	0.26
2014	30	14 432	0.18	0.19	0.15	0.13	0.21	0.29

5 SHOWCASES FOR THE COPYCAT LIBRARY

Within two showcases, we demonstrate the CopyCat software library: (1) deduplicating the qrel and run files of the TREC Web tracks using the lossless S_3 similarity measure, and (2) investigating whether relevance judgments from TREC tracks can be "transferred" to newer crawls by identifying near-duplicates.

Deduplication of Qrel and Run Files. The CopyCat software library can analyze standard TREC qrel and run files using ChatNoir [6] or Anserini [27] indices for random access to documents. As qrel and run files are typically small, they can be deduplicated using more expensive similarity measures than SimHash.

We deduplicate the qrel files (relevance judgments) and all run files of the TREC Web tracks with the lossless S_3 similarity at the previously determined threshold of 0.82. Table 4 shows the ratios of identified near-duplicates. The relevance judgments contain 14 % to 19 % near-duplicates. Note that mostly a higher relative ratio can be observed for the relevant documents while the "irrelevant" near-duplicates account for higher absolute numbers (not reported in the table). An average of 11–21 % near-duplicates within the top-10 ranks of runs submitted to the Web tracks further highlights the need for systematic deduplication—an effect that even increases when more documents from the rankings are considered (e.g., the top-100 or top-1000).

We pick document pairs from the top-1000 ranks of run files with an S_3 score above our threshold of 0.82 as the ground truth for a final sanity check. Table 5 shows the precision and recall of the CopyCat near-duplicates, confirming the high precision of 0.93 for our approach. Less than 1 % of ClueWeb09 documents have canonical links, which causes the low recall of SimHash with word-1-grams in equivalence classes of canonical links. Still, these canonical link duplicates contribute to the overall CopyCat recall with 0.03 to a total recall of 0.36.

Transferring Relevance Judgments. Relevance judgments for documents on a set of topics form the basis of evaluation in many shared tasks in information retrieval. For research on web search, shared tasks periodically exchange older crawls with newer ones to ensure timeliness. As the web is dynamic and evolves at a rapid pace, so do the web crawls, and the pages having been judged for older crawls may have disappeared. Hence, newer crawls often come with few or no available judgments at all. Already the move from the ClueWeb09 to the ClueWeb12 deprecated 73,883 relevance judgments for 198 Web track topics, which originally resulted from tedious manual labor of about 4–8 full-time person months (assuming 40-hour weeks with 30–60 seconds per judgment [26]).

Table 5: Precision and recall of near-duplicates in the CopyCat dataset ($S_3 \geq 0.82$) for runs submitted to the TREC web tracks at depth 1 000.

Method	All		Relevant		Irrelevant	
	Prec.	Rec.	Prec.	Rec.	Prec.	Rec.
SimHash duplicates	0.95	0.33	1.00	0.49	0.93	0.29
Canonical Link duplicates	0.90	0.08	0.99	0.17	0.79	0.11
CopyCat duplicates	0.93	0.36	0.99	0.54	0.87	0.34

To mitigate this regularly incurred loss when abandoning a well-annotated resource in favor of more recent ones, we evaluate to what extent near-duplicate detection may help to transfer relevance labels from the TREC Web track from the ClueWeb09 to the ClueWeb12 or the Common Crawl 2015-11. As for the transfer of relevance judgments between crawls, we identify web pages in the newer version that duplicate judged documents in the older one. Using the CopyCat software library, a robust two-stage pipeline is built. In a recall-oriented first step, candidate-pairs are collected using SimHash or, if applicable, with the help of canonical URLs. Using word 1-grams as SimHash document vector representations ensures a very good candidate recall (cf. Table 3) is ensured via the following process: (1) calculating fingerprints for all documents, (2) partitioning the fingerprints, (3) pruning partitions containing no judged documents, and (4) calculating the Hamming distances between fingerprints within each partition. In a final postprocessing, double-checking each candidate pair within a partition by calculating their S_3 scores on the raw documents and ignoring pairs below the previously set threshold of 0.82 ensures a high precision of the final near-duplicate sets.

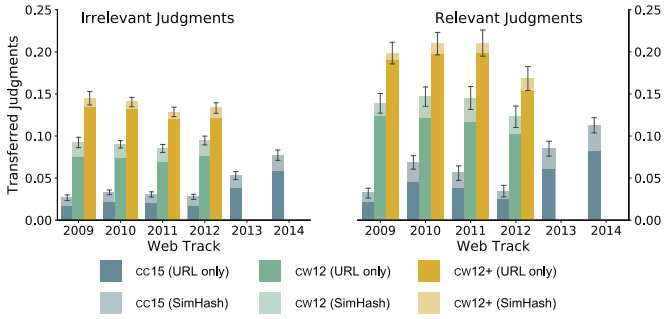
Tables 6a and b provide an overview of the transferred relevance judgments. We use the duplicate-free relevance judgments from the case study on near-duplicates in qrels to prevent counting any document more than once. Since the ClueWeb12 did not use previously judged ClueWeb09 documents as crawling seeds, we simulate that by enriching the ClueWeb12 to a ClueWeb12+ dataset via including snapshots of judged ClueWeb09 documents from the 2012 crawling period of the ClueWeb12 if they are available in the Wayback Machine.[8] Including such a "changed" crawling strategy in our analysis may impact decisions when today's shared tasks want to move from one crawl to a newer version.

The ratio of successfully transferred judgments is given per track and target crawl, each for judgments transferred based on their URLs only and with additional SimHashing. In total, 10 % of the ClueWeb09 relevance judgments are transferred to the ClueWeb12, and 3 % to the Common Crawl 2015-11. About 8 % of the ClueWeb12 judgments are transferred to the Common Crawl 2015-11. Leveraging the SimHash candidates allows for an additional 1.8 percentage points (ClueWeb12), and 1.4 points (Common Crawl 2015-11) on top of URL candidates only. Adding URLs for judged documents to the URL seeds of the ClueWeb12—as simulated in our ClueWeb12+ analysis—increases the number of transferable judgments by at

[8] https://archive.org/web/

Table 6: Ratio of successfully transferred (a) relevant and (b) irrelevant judgements per target crawl and transfer method with (c) resulting numbers of sparse topics, and (d) changes in the system rankings (Kendall's τ for sparse nDCG) on the ClueWeb12 (cw12), ClueWeb12+ (cw12+), and the Common Crawl 2015-11 (cc15).

(a) (b)



(c)

	Track	Sparse topics		
		cw12	cw12+	cc15
cw09	2009	39	46	11
cw09	2010	46	47	32
cw09	2011	36	40	15
cw09	2012	41	45	12
cw12	2013	–	–	29
cw12	2014	–	–	39

(d)

	Track	τ (nDCG)		
		cw12	cw12+	cc15
cw09	2009	0.82	0.83	0.73
cw09	2010	0.80	0.86	0.78
cw09	2011	0.85	0.86	0.83
cw09	2012	0.79	0.84	0.58
cw12	2013	-	-	0.89
cw12	2014	-	-	0.72

least 5 percentage points. Although this certainly is a lower bound (the Wayback Machine is not complete), most of the TREC NIST assessors' efforts can be considered "lost" for newer crawls.

Unfortunately, the small ratio of transferred judgments indicates that hardly any of the previous TREC Web track topics can only be reused with evaluation measures intended for dense judgments when the judgment process would be repeated. Experiments from the recent TREC Deep Learning track, on the other hand, suggest that evaluations based on sparsely judged topics with only few judgments at least correlate with the densely judged topics [10]. Table 6c shows the number of sparsely judged topics that can be transferred. To be included in the list, a sparsely judged topic has to have transferred at least one relevant and one irrelevant label and at least ten judgments overall. The largest number of such sparsely judged topics are retained for the ClueWeb12+ crawl (178 topics), followed by the original ClueWeb12 (162 topics), and the Common Crawl 2015-11 (138 topics, mainly from the ClueWeb12).

As an additional analysis, we test whether the transferred topics still help to distinguish between retrieval systems in terms of effectiveness. A number of simulated comparisons are conducted on the runs submitted to the Web tracks between 2009 and 2014. To simulate retrieval on the new crawls, the original run file's document IDs are mapped to their corresponding IDs in the new crawl, if a relevance judgment can be transferred, or else marked as unjudged by assigning a non-existing ID. Since the number of relevance judgments that could be transferred is so low, the average number of (transfer-)judged documents in the top-10 ranks of the submitted runs is only 2. Due to this small coverage, we use a sparse version of nDCG [24] that removes unjudged documents from the ranking and is known to perform well on sparse judgments.

Table 6d shows changes in the system rankings caused by the relevance transfer. We produce (sparse) nDCG system rankings with `trectools` [23] on both the topics with the original ClueWeb09 or ClueWeb12 dataset and the later datasets with transferred relevance judgments. The changes in the rankings are reported using Kendall's τ (1 indicates perfect, 0 random, and -1 perfect inverse agreement) and basically follow the number of transferred relevance judgments: ClueWeb12+ shows the highest correlation (between 0.83 and 0.86), followed by ClueWeb12 (between 0.79 and 0.85) and Common Crawl 2015-11 (between 0.58 and 0.89). The overall high correlation observed for the ClueWeb12+ suggests that including judged documents in the URL seeds of future crawls may indeed improve the longevity of the judgments. Unfortunately, we also find that for the top-5 systems, the correlation is almost entirely random ($\tau \in [0.11, 0.71]$). Despite the otherwise carefully optimistic conclusions, this confirms that the loss of over 80 % of all relevance judgments during the transfer cannot be easily compensated.

6 CONCLUSION AND FUTURE WORK

With the CopyCat resource, we provide lists of near-duplicates in the commonly used ClueWeb and Common Crawl datasets and a software toolkit to conduct deduplication on arbitrary datasets (e.g., TREC track runs). A straightforward use case of CopyCat are more robust retrieval system runs (without near-duplicates in their results) at a very low deduplication overhead on a researcher's end.

As a non-traditional application of the CopyCat resource, we examined whether documents judged by NIST assessors for the TREC Web tracks have near-duplicates in later crawls. Addressing the general transferability of TREC topics initially created for the ClueWeb09 and the ClueWeb12, we actually find only rather few near-duplicates of judged documents in newer crawls. Even though a certain amount of judgments can be "saved", new relevance judgments are definitely needed when previously used topics are to be reused in shared tasks on newer crawls.

CopyCat focuses on a precision-oriented near-duplicate detection such that one can be pretty sure that the returned output actually are near-duplicates. To further increase the recall, two directions are promising: (1) a more effective main content extraction such that near-duplicates can be better detected on the actual "retrieval-relevant" part of documents, and (2) dynamic content rendering for a more accurate representation of a web page.

Another interesting prospect for future work is to go beyond page-level granularity and to consider near-duplicates of relevant passages ("information nuggets") between documents. Estimating to what extent at least the important passages of documents judged as relevant can be identified in some other documents of a new crawl might be a promising way out of the the rather low document-level transferability of judged documents.

REFERENCES

[1] Eytan Adar, Jaime Teevan, Susan T. Dumais, and Jonathan L. Elsas. 2009. The Web Changes Everything: Understanding the Dynamics of Web Content. In *Proceedings of the Second International Conference on Web Search and Web Data Mining, WSDM 2009, Barcelona, Spain, February 9-11, 2009*. 282–291. https://doi.org/10.1145/1498759.1498837

[2] Amit Agarwal, Hema Swetha Koppula, Krishna P. Leela, Krishna Prasad Chitrapura, Sachin Garg, Pavan Kumar GM, Chittaranjan Haty, Anirban Roy, and Amit Sasturkar. 2009. URL Normalization for De-Duplication of Web Pages. In *Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM 2009, Hong Kong, China, November 2-6, 2009*. 1987–1990. https://doi.org/10.1145/1645953.1646283

[3] Bassma Alsulami, Maysoon Abulkhair, and Fathy Eassa. 2012. Near Duplicate Document Detection Survey. *International Journal of Computer Science and Communications Networks* 2, 2 (2012), 147–151.

[4] Yaniv Bernstein and Justin Zobel. 2004. A Scalable System for Identifying Co-derivative Documents. In *String Processing and Information Retrieval, 11th International Conference, SPIRE 2004, Padova, Italy, October 5-8, 2004, Proceedings*. 55–67. https://doi.org/10.1007/978-3-540-30213-1_6

[5] Yaniv Bernstein and Justin Zobel. 2005. Redundant Documents and Search Effectiveness. In *Proceedings of the 2005 ACM CIKM International Conference on Information and Knowledge Management, Bremen, Germany, October 31 - November 5, 2005*. 736–743. https://doi.org/10.1145/1099554.1099733

[6] Janek Bevendorff, Benno Stein, Matthias Hagen, and Martin Potthast. 2018. Elastic ChatNoir: Search Engine for the ClueWeb and the Common Crawl. In *Advances in Information Retrieval - 40th European Conference on IR Research, ECIR 2018, Grenoble, France, March 26-29, 2018, Proceedings (Lecture Notes in Computer Science, Vol. 10772)*, Gabriella Pasi, Benjamin Piwowarski, Leif Azzopardi, and Allan Hanbury (Eds.). Springer, 820–824. https://doi.org/10.1007/978-3-319-76941-7_83

[7] Andrei Z. Broder. 1997. On the Resemblance and Containment of Documents. In *Compression and Complexity of SEQUENCES 1997, Positano, Amalfitan Coast, Salerno, Italy, June 11-13, 1997, Proceedings*. 21–29. https://doi.org/10.1109/SEQUEN.1997.666900

[8] Moses Charikar. 2002. Similarity Estimation Techniques from Rounding Algorithms. In *Proceedings on 34th Annual ACM Symposium on Theory of Computing, May 19-21, 2002, Montréal, Québec, Canada*. 380–388. https://doi.org/10.1145/509907.509965

[9] Junghoo Cho and Hector Garcia-Molina. 2000. The Evolution of the Web and Implications for an Incremental Crawler. In *VLDB 2000, Proceedings of 26th International Conference on Very Large Data Bases, September 10-14, 2000, Cairo, Egypt*. 200–209. http://www.vldb.org/conf/2000/P200.pdf

[10] Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. Overview of the TREC 2019 Deep Learning Track. *CoRR* abs/2003.07820 (2020). arXiv:2003.07820 https://arxiv.org/abs/2003.07820

[11] Dennis Fetterly, Mark S. Manasse, and Marc Najork. 2003. On the Evolution of Clusters of Near-Duplicate Web Pages. In *1st Latin American Web Congress (LA-WEB 2003), Empowering Our Web, 10-12 November 2003, Sanitago, Chile*. 37–45. https://doi.org/10.1109/LAWEB.2003.1250280

[12] Dennis Fetterly, Mark S. Manasse, Marc Najork, and Janet L. Wiener. 2003. A Large-Scale Study of the Evolution of Web Pages. In *Proceedings of the Twelfth International World Wide Web Conference, WWW 2003, Budapest, Hungary, May 20-24, 2003*. 669–678. https://doi.org/10.1145/775152.775246

[13] Maik Fröbe, Janek Bevendorff, Jan Heinrich Reimer, Martin Potthast, and Matthias Hagen. 2020. Sampling Bias Due to Near-Duplicates in Learning to Rank. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, Jimmy Huang, Yi Chang, Xueqi Cheng, Jaap Kamps, Vanessa Murdock, Ji-Rong Wen, and Yiqun Liu (Eds.). ACM, 1997–2000. https://doi.org/10.1145/3397271.3401212

[14] Maik Fröbe, Jan Philipp Bittner, Martin Potthast, and Matthias Hagen. 2020. The Effect of Content-Equivalent Near-Duplicates on the Evaluation of Search Engines. In *Advances in Information Retrieval - 42nd European Conference on IR Research, ECIR 2020, Lisbon, Portugal, April 14-17, 2020, Proceedings, Part II (Lecture Notes in Computer Science, Vol. 12036)*, Joemon M. Jose, Emine Yilmaz, João Magalhães, Pablo Castells, Nicola Ferro, Mário J. Silva, and Flávio Martins (Eds.). Springer, 12–19. https://doi.org/10.1007/978-3-030-45442-5_2

[15] Monika Rauch Henzinger. 2006. Finding Near-Duplicate Web Pages: A Large-Scale Evaluation of Algorithms. In *SIGIR 2006: Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, Seattle, Washington, USA, August 6-11, 2006*. 284–291. https://doi.org/10.1145/1148170.1148222

[16] Sung Jin Kim and Sang Ho Lee. 2005. An Empirical Study on the Change of Web Pages. In *Web Technologies Research and Development - APWeb 2005, 7th Asia-Pacific Web Conference, Shanghai, China, March 29 - April 1, 2005, Proceedings*. 632–642. https://doi.org/10.1007/978-3-540-31849-1_62

[17] Christian Kohlschütter, Peter Fankhauser, and Wolfgang Nejdl. 2010. Boilerplate Detection Using Shallow Text Features. In *Proceedings of the Third International Conference on Web Search and Web Data Mining, WSDM 2010, New York, NY, USA, February 4-6, 2010*, Brian D. Davison, Torsten Suel, Nick Craswell, and Bing Liu (Eds.). ACM, 441–450. https://doi.org/10.1145/1718487.1718542

[18] Hema Swetha Koppula, Krishna P. Leela, Amit Agarwal, Krishna Prasad Chitrapura, Sachin Garg, and Amit Sasturkar. 2010. Learning URL Patterns for Webpage De-Duplication. In *Proceedings of the Third International Conference on Web Search and Web Data Mining, WSDM 2010, New York, NY, USA, February 4-6, 2010*, Brian D. Davison, Torsten Suel, Nick Craswell, and Bing Liu (Eds.). ACM, 381–390. https://doi.org/10.1145/1718487.1718535

[19] Jimmy Lin and Peilin Yang. 2019. The Impact of Score Ties on Repeatability in Document Ranking. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*. 1125–1128. https://doi.org/10.1145/3331184.3331339

[20] Gurmeet Singh Manku, Arvind Jain, and Anish Das Sarma. 2007. Detecting Near-Duplicates for Web Crawling. In *Proceedings of the 16th International Conference on World Wide Web, WWW 2007, Banff, Alberta, Canada, May 8-12, 2007*, Carey L. Williamson, Mary Ellen Zurko, Peter F. Patel-Schneider, and Prashant J. Shenoy (Eds.). ACM, 141–150. https://doi.org/10.1145/1242572.1242592

[21] Alexandros Ntoulas, Junghoo Cho, and Christopher Olston. 2004. What's new on the Web?: The Evolution of the Web from a Search Engine Perspective. In *Proceedings of the 13th international conference on World Wide Web, WWW 2004, New York, NY, USA, May 17-20, 2004*, Stuart I. Feldman, Mike Uretsky, Marc Najork, and Craig E. Wills (Eds.). ACM, 1–12. https://doi.org/10.1145/988672.988674

[22] Christopher Olston and Sandeep Pandey. 2008. Recrawl Scheduling Based on Information Longevity. In *Proceedings of the 17th International Conference on World Wide Web, WWW 2008, Beijing, China, April 21-25, 2008*, Jinpeng Huai, Robin Chen, Hsiao-Wuen Hon, Yunhao Liu, Wei-Ying Ma, Andrew Tomkins, and Xiaodong Zhang (Eds.). ACM, 437–446. https://doi.org/10.1145/1367497.1367557

[23] João R. M. Palotti, Harrisen Scells, and Guido Zuccon. 2019. TrecTools: An Open-source Python Library for Information Retrieval Practitioners Involved in TREC-like Campaigns. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, Paris, France, July 21-25, 2019*, Benjamin Piwowarski, Max Chevalier, Éric Gaussier, Yoelle Maarek, Jian-Yun Nie, and Falk Scholer (Eds.). ACM, 1325–1328. https://doi.org/10.1145/3331184.3331399

[24] Tetsuya Sakai. 2007. Alternatives to Bpref. In *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*, Wessel Kraaij, Arjen P. de Vries, Charles L. A. Clarke, Norbert Fuhr, and Noriko Kando (Eds.). ACM, 71–78. https://doi.org/10.1145/1277741.1277756

[25] Gilles Vandewiele, Isabelle Dehaene, György Kovács, Lucas Sterckx, Olivier Janssens, Femke Ongenae, Femke De Backere, Filip De Turck, Kristien Roelens, Johan Decruyenaere, Sofie Van Hoecke, and Thomas Demeester. 2021. Overly Optimistic Prediction Results on Imbalanced Data: A Case Study of Flaws and Benefits when Applying Over-Sampling. *Artif. Intell. Medicine* 111 (2021), 101987. https://doi.org/10.1016/j.artmed.2020.101987

[26] Ellen M. Voorhees. 2001. The Philosophy of Information Retrieval Evaluation. In *Evaluation of Cross-Language Information Retrieval Systems, Second Workshop of the Cross-Language Evaluation Forum, CLEF 2001, Darmstadt, Germany, September 3-4, 2001, Revised Papers (Lecture Notes in Computer Science, Vol. 2406)*, Carol Peters, Martin Braschler, Julio Gonzalo, and Michael Kluck (Eds.). Springer, 355–370. https://doi.org/10.1007/3-540-45691-0_34

[27] Peilin Yang, Hui Fang, and Jimmy Lin. 2017. Anserini: Enabling the Use of Lucene for Information Retrieval Research. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval, Shinjuku, Tokyo, Japan, August 7-11, 2017*, Noriko Kando, Tetsuya Sakai, Hideo Joho, Hang Li, Arjen P. de Vries, and Ryen W. White (Eds.). ACM, 1253–1256. https://doi.org/10.1145/3077136.3080721