# The Open Web Index

## Crawling and Indexing the Web for Public Use

Gijs Hendriksen[1], Michael Dinzinger[2], Sheikh Mastura Farzana[3], Noor Afshan Fathima[4], Maik Fröbe[5], Sebastian Schmidt[6], Saber Zerhoudi[2], Michael Granitzer[2], Matthias Hagen[5], Djoerd Hiemstra[1], Martin Potthast[6,7], and Benno Stein[8]

[1] Radboud University
[2] University of Passau
[3] German Aerospace Center (DLR)
[4] CERN
[5] Friedrich-Schiller-Universität Jena
[6] Leipzig University
[7] ScaDS.AI
[8] Bauhaus-Universität Weimar

**Abstract** Only few search engines index the Web at scale. Third parties who want to develop downstream applications based on web search fully depend on the terms and conditions of the few vendors. The public availability of the large-scale Common Crawl does not alleviate the situation, as it is often cheaper to crawl and index only a smaller collection focused on a downstream application scenario than to build and maintain an index for a general collection the size of the Common Crawl. Our goal is to improve this situation by developing the *Open Web Index*.
The Open Web Index is a publicly funded basic infrastructure from which downstream applications will be able to select and compile custom indexes in a simple and transparent way. Our goal is to establish the Open Web Index along with associated data products as a new open web information intermediary. In this paper, we present our first prototype for the Open Web Index and our plans for future developments. In addition to the conceptual and technical background, we discuss how the information retrieval community can benefit from and contribute to the Open Web Index—for example, by providing resources, by providing pre-processing components and pipelines, or by creating new kinds of vertical search engines and test collections.

## 1 Introduction

Web search is an important technology for accessing the information on the Web. However, operating a full-scale web search engine is far from trivial. Crawling, processing, and indexing the Web consumes a large amount of resources, even without factoring in the large volume of queries that a search engine might have to process. As a result, only a handful of large corporations have been able to develop and operate commercial search engines, and they currently dominate

the search engine market. This is in contrast to the recent release of data sets and open source models for generative AI. While companies like OpenAI made rapid progress on large language models and commercialized their successes, withholding details about their training data, model infrastructures, and training methods, the open source community quickly caught up. Currently, dozens of open-source AI models rival the effectiveness of closed-source models [21, 23]. Despite many efforts in previous years, this has not been the case with web search. The few alternative search engines that have emerged do not share their data, index, or other details.

We introduce the Open Web Index, the first collaborative and federated data structure for crawling, enriching, and indexing the Web, distributed across multiple European data centers. The Open Web Index is inspired by Lewandowski's idea of an "open web index" [13] and corresponding core principles [7]. However, we not only provide access via an API, but treat the entire index and associated data products as open data. Furthermore, we enrich the crawled web content with a variety of metadata that can in turn drive *vertical* search engines. By publishing the index itself, we enable a new landscape of search engines, where each vertical can target different audiences based on tailored ranking strategies that meet their respective values (e.g., sustainability or privacy). In addition, the Open Web Index enables the training of specialized language models on different subsets of the Web. Our goal is to gain traction in the information retrieval and open source communities, allowing interested parties to contribute to the Open Web Index. This may include new content analysis modules for the preprocessing pipeline and evaluation components for the open evaluation framework.

While the development of the index is part of the ongoing Open Web Search research project[9], the crawling, preprocessing, and indexing pipelines already run on two European data centers producing ca. 1TB of data per day and location. The current activities focus on three main areas: (1) conducting further research on the analysis of large web collections to expand the metadata provided in the index files, (2) implementation of the three major pipeline steps and onboarding of additional data centers, and (3) fostering the open source and research community around the Open Web Index.

## 2 Related Work

The idea of collecting data on a massive scale for various purposes is not new. Several projects in the past have engaged in such endeavors. Most notably, the Common Crawl project[10] stands out as a significant effort in this direction. The Common Crawl initiative collects data from the Web at large scale and makes it accessible to the public. Several derivative projects, such as C4 [20], the Pile [4], of Web Data Commons [18], have built upon the resources provided by Common Crawl, indicating its importance and far-reaching benefits in the community.

[9] https://openwebsearch.eu/
[10] https://commoncrawl.org/

Another related project, LAION,[11] takes a similar approach. The non-profit organization provides data sets, tools and models in order to strengthen open source machine learning research. Furthermore, Curlie,[12] which was previously known as the Open Directory Project (ODP) and DMOZ, offers a manually curated directory of the Web. By fully relying on the power of human editors, Curlie stands out due to its elaborate and qualitatively advanced approach to data categorization and authenticity.

However, our efforts are not only focused on data collection, but also on web search applications based on the crawled and enriched data. Aside from the few major search engines (among others, Google, Bing and Yandex), several alternative search engines have emerged over the years from which, in particular, DuckDuckGo, Ecosia, and Startpage.com became popular. They do not operate their own crawling and indexing infrastructure, but make use of a search API offered by Bing or Google. Other search engines, such as Mojeek and Qwant, have tried to present themselves as viable alternatives to the large commercial search engines by building their own index. Both providers are particularly committed to preserving their neutrality and user privacy. Although they have not been able to create and share their indexes with the public, their efforts underscore the need and necessity for more players in the search engine market. Yet, both Mojeek's[13] and Qwant's[14] index is only a fraction as large as that of the market leader Google. Qwant currently still relies on Bing's index to supplement its own index. We believe that a collaborative crawling and indexing effort can help make the Open Web Index a good alternative to the current gatekeepers' search indexes—both in terms of scope and quality.

Recent trends have highlighted the importance of personalized search experiences, where search engines strive to understand user intent and context to provide more tailored results. Additionally, the deployment of natural language understanding and conversational agents in search engines has transformed the way users interact with online information. However, the development of specialized search engines depends on the availability of specially curated data. The Open Web Index aims to support these needs and will offer search engine developers different types of curated indexes in different sizes.

In addition to the technical aspects, ethical considerations and responsible data handling are critical aspects that are increasingly becoming the focus of data collection initiatives. Recent regulations and discussions surrounding data privacy, such as the General Data Protection Regulation (GDPR) in Europe and the California Consumer Privacy Act (CCPA), reflect the global shift towards safeguarding individual privacy. In our initiative, we intend to respect all aspects of data privacy and ethics while crawling, storing and distributing web data.

[11] https://laion.ai/

[12] https://curlie.org

[13] https://blog.mojeek.com/2022/03/five-billion-pages.html

[14] https://betterweb.qwant.com/en/2023/09/18/web-indexing-where-is-qwants-independence/

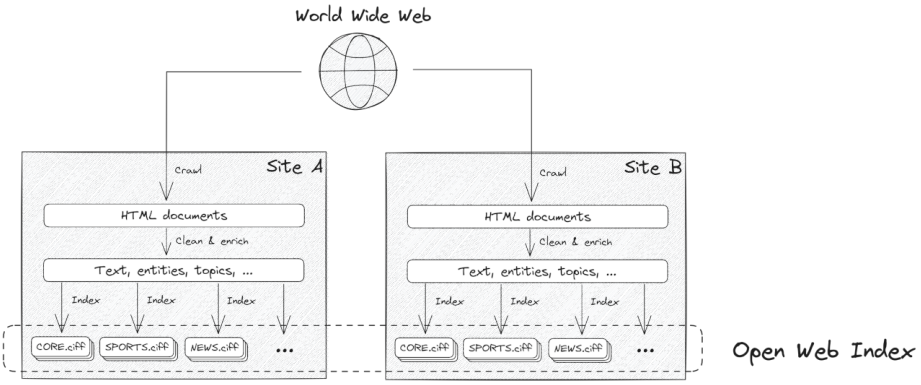


**Figure 1.** Distributed pipeline architecture of the Open Web Index. Each data center is responsible for crawling, cleaning, enriching, and indexing its own subset of the web. Which documents are crawled by which data center is decided by the common frontier service. The definition and creation of index verticals and the choice of which enrichments to include are still open research questions.

## 3 Infrastructure of the Open Web Index

Figure 1 shows an overview of the Open Web Index architecture. Particularly noticeable are the improvements in accessing current search indexes compared to Lewandowski's [13] API-based proposal. Our federated data infrastructure crawls, enriches, and indexes web content in a distributed manner across multiple European data centers. The resulting indexes are divided into a set of predefined, possibly overlapping, verticals and are continuously updated over time. In addition to these vertical indexes, we also create "core indexes" that contain subsets of highly frequented or otherwise important websites. The underlying rationale is that a relatively small subset of the Web can already answer a large majority of queries. For example, Goel et al. [5] have shown that a subset of only 10,000 domains was responsible for about 80% of users' clicks in 2010, and, more recently, the creation of the ClueWeb22 corpus [19] confirms that a small number of domains still accounts for a large share of user clicks. In addition to the core index of highly frequented domains, other, use-case-specific indexes can be constructed. Examples include the user-curated collection Curlie[15] and the set of all Wikipedia pages with their outgoing links. All indexes are stored in Common Index File Format (CIFF) [15] for ease of transfer and compatibility with existing software. The remainder of this section explains the details of the various components or layers of our architecture. All software components of this architecture are open source as well as archived in our Zenodo community.[16][17]

[15]https://curlie.org
[16]https://opencode.it4i.eu/openwebsearcheu-public/
[17]https://zenodo.org/communities/owseu/

### 3.1 Crawling

A central pillar of the Open Web Index is the crawling pipeline, which is designed to seamlessly capture large volumes of web data. The pipeline relies on a distributed architecture, where multiple clusters spread across several European data centers jointly retrieve and extract web content. These nodes are coordinated from a central hub, the Open Web Index Frontier, to ensure orchestration and efficient data collection. The Open Web Index Frontier is built on the open source URLFrontier project, which implements plug-and-play functionality for new crawling agents.[18] Since all communication between remote software components is based on the Frontier, new crawling agents can be easily integrated into the live system. Each link in the Frontier can be pre-categorized based on various parameters (e.g., topic, license, genre) to ensure that the collected data is organized and ready for downstream applications. Furthermore, the Frontier manages the different node resources to ensure efficient load balancing across the federated nodes.

To operationalize our crawling pipeline, we introduce a re-crawl mechanism specifically for Common Crawl dumps. Unlike Common Crawl, we crawl on a daily basis and save the results as WARC (Web ARChive) files.[19] This approach allows us to provide index delta files and supplemental data products at daily intervals, as opposed to monthly or bi-monthly data dumps. In addition, each link goes through a rigorous filtering process supported by an exclusion list mechanism. This list is constantly updated with malicious URLs, similar to platforms such as URLHaus.[20] This strategy ensures the highest quality of data.

By prioritizing certain top-level domains in our crawl exploration as well as using content negotiation headers and the IP locations of the various European data centers from which the crawls are conducted, we aim to collect web resources in many different languages. The goal is to increase the language diversity in the Open Web Index and thereby reduce the existing linguistic bias in downstream tasks towards high-resource languages such as English [12].

Our initial crawling runs across all our data centers resulted in a cumulative crawling rate of over 30 million web pages per day. Currently, our comprehensive crawling generates more than 1 terabyte of data daily.

**Crawling on Demand** Given the significant cost and resources required for crawling or scraping the Web, we introduce "crawling on demand". This allows authorized users to initiate a customized crawling process tailored to their specific needs. Users provide a curated list of seed URLs. Upon receiving this list, our system starts a dockerized crawl cluster tailored for the task at hand. Communicating with the Frontier service, the crawler detects whether a URL has already been crawled. If it has, its content is retrieved without unnecessary re-crawling, ensuring efficient resource usage. Users receive a daily update of WARC

[18] http://urlfrontier.net
[19] https://iipc.github.io/warc-specifications/
[20] https://urlhaus.abuse.ch/

files corresponding to their seed URLs, ensuring access to the freshest and most relevant data. In addition, we are nearing completion of the addition of "index on demand" to this service. This service not only initiates the cleaning and enrichment of the crawled documents, but also delivers the data as index files for immediate integration into search engines.

**Web publisher controls** As part of our crawling, we aim to improve the usage control facilities for web publishers. With the advent of generative AI applications, many content creators are looking for ways to better protect their publicly available web assets and opt out of their use for text and data mining or AI training. These opt-out signals are conveyed in various forms, including the robots exclusion protocol ("robots.txt") [11], meta tags such as "`noml`",[21] and emerging web standards such as the TDM Reservation Protocol.[22] We evaluate and incorporate these machine-readable signals in our crawlers, propagating publisher usage preferences to downstream users of the Open Web Index. Additionally, we adhere to established politeness policies, such as crawling intervals communicated by site managers through robots.txt.

### 3.2 Preprocessing

The preprocessing pipeline extracts various types of page-level metadata in addition to the cleaned text from the WARC files created during daily crawling activities. The extracted data is provided in Apache Parquet file format as part of the Open Web Index and can be used to enrich index files, as described in more detail in subsection 4.1.

Following the daily crawling activities, the preprocessing is executed as daily batch jobs at the different data centers storing the WARC files. First, an Apache Spark cluster is created on the respective HPC cluster using the Magpie script collection.[23] Then, the preprocessing job is submitted to the newly created Spark cluster and the extracted metadata is saved as Parquet files. Currently, the preprocessing pipeline extracts the plain text from the HTML code of each page, as well as various information from the WARC and HTTP headers and URL components. In addition, two types of metadata are created to enable partitioning of the index files: the language of the document and, if available, a label for the domain based on the labels collected by the Curlie community [16]. We plan to incorporate more metadata into the preprocessing pipeline throughout the duration of the Open Web Search project.

**Evaluation benchmarks** Since the preprocessing pipeline is built on a modular architecture, it allows the integration of content analysis modules developed by third parties. This will help expand the amount of metadata extracted

[21] https://noml.info
[22] https://www.w3.org/2022/tdmrep/
[23] https://github.com/LLNL/magpie/

from crawls with the help of the open-source community. To ensure both sufficient quality and throughput of content analysis modules, new modules will be evaluated in a dedicated evaluation layer. This layer runs on the TIRA framework [2, 3], which provides the means for evaluation as a service focused on information retrieval research. TIRA can host shared tasks on a given research problem and executes submitted software in sandbox machines without internet access to improve reproducibility.

Each new candidate module is evaluated against problem-specific benchmarking data, such as a set of labeled data for classification tasks. For tasks not previously included in the preprocessing pipeline, benchmarking data must be provided by the party developing the module. We also work on increasing the number of benchmarking datasets and submitted modules for a given task to support the development of high quality content analysis modules.

### 3.3 Indexing

The indexer takes the cleaned text from the preprocessing pipeline and converts it to a full-text index. The index is partitioned into a series of shards using various metadata values. Currently, each combination of top-level range, language, and Curlie topic is assigned to a different shard. However, determining the optimal metadata set for partitioning the index is still an open research question, and we will evaluate different approaches during the development of Open Web Index.

Each shard is a separate CIFF index [15] and can be easily downloaded along with the Parquet files containing the relevant metadata and clean text. A downstream search engine or end user can select any combination of these shards to create a custom vertical search application—public, commercial, or personal.

Similar to the preprocessing pipeline, indexing is performed as a daily Spark batch job that runs after all content for a day has been preprocessed. Magpie is used to provision the Spark cluster within an HPC allocation, after which the indexer is executed.

### 3.4 Challenges

While we have elaborately discussed the importance and necessity of an Open Web Index, there still exist crucial challenges regarding our proposed federated infrastructure. A major challenge to consider is the sustainability of the Open Web Index in the long term. The Open Web Search project is currently nurtured by a diverse team representing various institutes and countries. However, such a publicly funded project is inherently limited in both time and resources.

Within the project consortium, we are already discussing ways in which we can ensure the index's sustainability in the future. Identifying responsible parties or entities tasked with maintaining the index in the long run is an important point in these discussions. As part of the sustainability of the Open Web Index, we also wish to integrate the open source and open data communities into its development, and are discussing how this effort can be coordinated. The steps

we take in these directions are crucial to guaranteeing the ongoing relevance and enduring presence of the Open Web Index beyond its initial phases.

Our federated infrastructure comes with additional challenges that any technical infrastructure has to tackle, such as security, hardware/software/service management, and agreements with end users through usage policies and service level agreements. These become especially difficult for a public project with limited resources, where no dedicated teams are in place to manage these issues. We are discussing and handling these challenges for our current infrastructure, and will consider ways in which they can be handled in the long term as well.

## 4 Usage of the Open Web Index

By publishing a comprehensive web index, we are supporting the information retrieval community in a very tangible way. The following section focuses on the data products provided as part of the Open Web Index. To further enhance the user experience, we also introduce an advanced concept to make parts of the index available to future developers of vertical search applications.

### 4.1 Data products

The Open Web Index consists of a set of data products that can be used by downstream search engines or other data-intensive applications. Depending on ethical considerations and to the extent legally possible, we make this data available for public download. Below, we discuss the different types of data outputs we plan to generate, how they are created, and what uses they may have.

**Index files** The first and most important goal of the Open Web Index is to enable downstream (vertical) search engines. To achieve this, we periodically distribute the index files of all crawled and cleaned content and provide the inverted files in CIFF format [15]. Several search engines, including JASSv2, Lucene (and thus Anserini and Solr), PISA, OldDog, and Terrier, already support the import of CIFF indexes. This makes it easy to develop a vertical search engine based on the Open Web Index with mature software and minimal effort.

In recent years, the use of dense or sparse embeddings for ranking has become very important in the information retrieval community. Therefore, an interesting avenue for the Open Web Index is the inclusion of embedding-based indexes alongside the term-based CIFF index that we already offer. This option will further facilitate the creation of downstream search engines by eliminating the need to re-compute the embeddings every time, contributing to "Green IR" [22]. However, given the size of the web, we need to ensure that the embeddings are useful and can be computed efficiently before running comprehensive embedding models for the entire collection. In our future work on this problem, we will consider both dense [10] and sparse embeddings [1].

**Clean text** A search engine also needs access to the cleaned text of the indexed documents in order to present the search results correctly. In classical search engines with a results page containing "ten blue links", the cleaned text is used to generate informative snippets that are relevant to the query. In conversational web search systems that apply retrieval-augmented generation [14] to summarize search results in natural language, the document content is needed to generate the full-text response to a query. To support such use cases, we plan to provide the plain text (along with document-level metadata) in the form of Parquet files.

Another important opportunity arising from the availability of clean text is the ability to (pre-)train large open source language models. Currently, this is typically done with public datasets, of which Common Crawl (including derived datasets such as C4 [20] and the Pile [4]) are the largest. Our cleaned text could provide an alternative to the Common Crawl's WET (Web Extracted Text) files. In addition, our enriched content (described in more detail below) will allow us to select subsets of web content for training smaller, more focused language models. For example, filtering German web pages could prove useful for training German language models, and focusing on scientific data could result in a language model that can be used more effectively in the scientific domain.

**Structured information** The web is full of structured information that can be used to build knowledge graphs and support many downstream applications. Similar to Web Data Commons [18] (derived from Common Crawl), we plan to extract and share entities from Schema.org [8] from various semantic annotations embedded in web pages. Furthermore, the textual content of the page may contain additional entities that are not included in machine-readable markup, but are still useful for various use cases. For example, the text could contain names, dates, or location information. To extract such mentions, we include the REBL Batch Entity Linker [9, 24] in our preprocessing pipeline. In addition, we implement and apply geoparsing tools that allow us to extract place names and place references and map them to physical geographic locations. Since there are currently no end-to-end geoparsers that can perform this task on the Web, we are using tools like Geoparsepy for inspiration [17].

**Page-level metadata** For the development of vertical search engines, we would like to assign web documents to meaningful index domains. To achieve this, we will collect different types of page-level features that can be useful for index partitioning, such as language, topic, and genre.

In addition, we will work on the development of an information nutrition label that includes benchmarks for the readability, factuality, and other aspects of any web document [6]. This will allow us to make judgments about the quality and trustworthiness of documents in the Open Web Index. In turn, this could enable search engine developers to create better ranking models, but more importantly, it allows end users to make an informed decision about which documents to access given a list of search results.

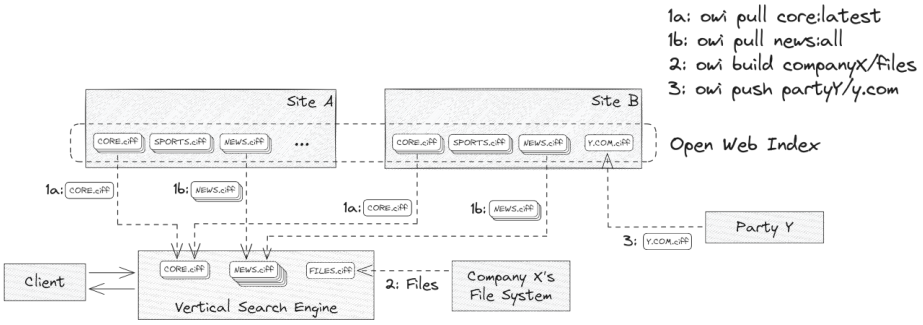


**Figure 2.** Interaction between a search engine and the Open Web Index. Downstream search engines that rely on the Open Web Index can (1) retrieve prebuilt indexes, (2) create their own indexes, and (3) push indexes for sharing with others.

To empower developers and end users, we will develop classifiers to identify textual content that may be upsetting or otherwise disturbing to certain audiences. Adding such trigger warnings as page-level metadata allows search engine providers to display them alongside search results, empowering end users to avoid potentially harmful content. The necessary components in our pre-processing pipeline are built on existing research on trigger warning classification [25].

### 4.2 Interaction with the Open Web Index

Since the Open Web Index is intended to be used by a variety of search engines, it must also be easy for search engine developers to use the index (or parts of it) for their own purposes. To this end, we envision the Open Web Index becoming a (distributed) information system that can be used in a manner similar to the well-known Docker Hub. However, instead of container images, the Open Web Index contains prebuilt indexes that are immediately usable. The Open Web Index enables several downstream applications, which are depicted in Figure 2:

1. Users or organizations can download (or 'pull') a specific, pre-built index.
   (a) They can choose a specific timestamp or checkpoint of the index (e.g. `:latest` for the most recent version).
   (b) They can choose to download a selection of checkpoints, instead of only a single one (e.g., `:all` for the complete history of a specific index).
2. Users or organizations can create (or 'build') their own index locally, using a data set of their choosing (e.g., privacy-sensitive data, such as a corporate filesystem, or personal email).
3. Users or organizations can upload (or 'push') a custom index or custom metadata to contribute to the Open Web Index.

With the creation of the Open Web Index as a data product rather than an API service, several challenges arise:

1. *Index merge:* The Open Web Index allows downloading (or 'pulling') web indexes for specific verticals. However, to support efficient retrieval, it may be necessary to merge these vertical indexes into a single usable index. Possible options for the index merging step include (1) no merging (pure federated search), (2) client-side merging, (3) server-side merging, and (4) a hybrid merging approach. Further research is needed to investigate the tradeoffs between efficiency and usability for each of these methods in order to make a decision on a method that is useful in practice.
2. *Freshness:* The Open Web Index and the indexes retrieved by downstream search engines need to be updated regularly to ensure that search results remain accurate. To accomplish this, we could (a) update an index incrementally by marking documents as obsolete and using a smaller, separate index for those updated documents, or (b) rebuild and replace the entire index from time to time. We will explore which method is best to ensure timeliness and also how we can extend these processes to also ensure the timeliness of downstream search engines. The federated structure of the Open Web Index also allows us to apply different "freshness" policies to specific sub-indexes, e.g., updating the news index more frequently compared to more static indexes.
3. *Index curation:* Eventually, the question arises how to ensure the quality of the contributed search indexes. One possible quality assurance model is to provide a small number of official and manually reviewed indexes. Similar to common practice in software repository platforms, users could also be given the option to star high-quality indexes provided by the community.

During the development of the Open Web Index, we continue to explore these issues and try to find solutions that make using the Open Web Index as easy and user-friendly as possible.

## 5 Conclusion

Inspired by recent advances in open-source AI models and Lewandowski's idea of an open web index, we introduce the Open Web Index. The main goal is to facilitate the development of search applications (e.g., to ground retrieval-augmented generation systems or to create vertical search engines for specific domains) without having to rely on the APIs of one of the few web-scale search engines typically operated by large corporations. We make the Open Web Index compatible with existing software by using the CIFF format and, in addition to the inverted files, we provide page-level metadata to support developers in customizing the selection of documents that meet the needs of their specific application. By using our 'crawling on demand' feature, developers can simply specify a list of URLs in order to receive the corresponding WARC files from either our existing crawls or from specially created new crawl clusters.

The Open Web Index is open for contributions from others. The IR community, for instance, can contribute by developing additional preprocessing components, by creating new kinds of vertical search engines and test collections, or by providing resources.

## Acknowledgments

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

## Bibliography

[1] Formal, T., Piwowarski, B., Clinchant, S.: SPLADE: Sparse Lexical and Expansion Model for First Stage Ranking. p. 2288–2292, SIGIR '21, Association for Computing Machinery, New York, NY, USA (2021), ISBN 9781450380379

[2] Fröbe, M., Reimer, J.H., MacAvaney, S., Deckers, N., Reich, S., Bevendorff, J., Stein, B., Hagen, M., Potthast, M.: The Information Retrieval Experiment Platform. In: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (2023)

[3] Fröbe, M., Wiegmann, M., Kolyada, N., Grahm, B., Elstner, T., Loebe, F., Hagen, M., Stein, B., Potthast, M.: Continuous Integration for Reproducible Shared Tasks with TIRA.io. In: Advances in Information Retrieval. 45th European Conference on IR Research (ECIR 2023), Lecture Notes in Computer Science, Springer (2023)

[4] Gao, L., Biderman, S., Black, S., Golding, L., Hoppe, T., Foster, C., Phang, J., He, H., Thite, A., Nabeshima, N., Presser, S., Leahy, C.: The Pile: An 800GB Dataset of Diverse Text for Language Modeling (Dec 2020)

[5] Goel, S., Broder, A.Z., Gabrilovich, E., Pang, B.: Anatomy of the Long Tail: Ordinary People with Extraordinary Tastes. In: Davison, B.D., Suel, T., Craswell, N., Liu, B. (eds.) Proceedings of the Third International Conference on Web Search and Web Data Mining, WSDM 2010, New York, NY, USA, February 4-6, 2010, pp. 201–210, ACM (2010)

[6] Gollub, T., Potthast, M., Stein, B.: Shaping the Information Nutrition Label. In: Albakour, D., Corney, D., Gonzalo, J., Martinez, M., Poblete, B., Valochas, A. (eds.) 2nd International Workshop on Recent Trends in News Information Retrieval (NewsIR 2018) at ECIR, CEUR Workshop Proceedings, vol. 2079, pp. 9–11 (Mar 2018), ISSN 1613-0073

[7] Granitzer, M., Voigt, S., et al.: Impact and Development of an Open Web Index for Open Web Search. Journal of the Association for Information Science and Technology (2023)

[8] Guha, R.V., Brickley, D., MacBeth, S.: Schema.org: Evolution of Structured Data on the Web: Big data makes common schemas even more necessary. Queue **13**(9), 10–37 (Nov 2015), ISSN 1542-7730

[9] Kamphuis, C., Hasibi, F., Lin, J., de Vries, A.P.: REBL: Entity Linking at Scale. In: Alonso, O., Baeza-Yates, R., King, T.H., Silvello, G. (eds.) Proceedings of the Third International Conference on Design of Experimental Search & Information REtrieval Systems, San Jose, CA, USA, August 30-31, 2022, CEUR Workshop Proceedings, vol. 3480, pp. 68–75, CEUR-WS.org (2022)

[10] Khattab, O., Zaharia, M.: ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, p. 39–48, SIGIR '20, Association for Computing Machinery, New York, NY, USA (2020), ISBN 9781450380164

[11] Koster, M., Illyes, G., Zeller, H., Sassman, L.: RFC 9309 Robots Exclusion Protocol (2022)
[12] Kreutzer, J., et al.: Quality at a Glance: An Audit of Web-Crawled Multilingual Datasets (2021)
[13] Lewandowski, D.: The Web Is Missing an Essential Part of Infrastructure: An Open Web Index. Commun. ACM **62**(4), 24 (2019)
[14] Li, H., Su, Y., Cai, D., Wang, Y., Liu, L.: A Survey on Retrieval-Augmented Text Generation. arXiv preprint arXiv:2202.01110 (2022)
[15] Lin, J., Mackenzie, J., Kamphuis, C., Macdonald, C., Mallia, A., Siedlaczek, M., Trotman, A., de Vries, A.: Supporting Interoperability Between Open-Source Search Engines with the Common Index File Format. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2149–2152 (2020)
[16] Lugeon, S., Piccardi, T.: Curlie Dataset - Language-agnostic Website Embedding and Classification (1 2023), https://doi.org/10.6084/m9.figshare.19406693.v5, URL https://figshare.com/articles/dataset/Curlie_Dataset_-_Language-agnostic_Website_Embedding_and_Classification/19406693
[17] Middleton, S.E., Kordopatis-Zilos, G., Papadopoulos, S., Kompatsiaris, Y.: Location Extraction from Social Media: Geoparsing, Location Disambiguation, and Geotagging. ACM Transactions on Information Systems (TOIS) **36**(4), 1–27 (2018)
[18] Mühleisen, H., Bizer, C.: Web Data Commons - Extracting Structured Data from Two Large Web Corpora. In: Bizer, C., Heath, T., Berners-Lee, T., Hausenblas, M. (eds.) WWW2012 Workshop on Linked Data on the Web, Lyon, France, 16 April, 2012, CEUR Workshop Proceedings, vol. 937, CEUR-WS.org (2012)
[19] Overwijk, A., Xiong, C., Liu, X., VandenBerg, C., Callan, J.: ClueWeb22: 10 Billion Web Documents with Visual and Semantic Information (Dec 2022)
[20] 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)
[21] Scao, T.L., Fan, A., Akiki, C., Pavlick, E., Ilic, S., Hesslow, D., Castagné, R., Luccioni, A.S., Yvon, F., Gallé, M., Tow, J., Rush, A.M., Biderman, S., Webson, A., Ammanamanchi, P.S., Wang, T., Sagot, B., Muennighoff, N., del Moral, A.V., Ruwase, O., Bawden, R., Bekman, S., McMillan-Major, A., Beltagy, I., Nguyen, H., Saulnier, L., Tan, S., Suarez, P.O., Sanh, V., Laurençon, H., Jernite, Y., Launay, J., Mitchell, M., Raffel, C., Gokaslan, A., Simhi, A., Soroa, A., Aji, A.F., Alfassy, A., Rogers, A., Nitzav, A.K., Xu, C., Mou, C., Emezue, C., Klamm, C., Leong, C., van Strien, D., Adelani, D.I., et al.: BLOOM: A 176B-Parameter Open-Access Multilingual Language Model. CoRR **abs/2211.05100** (2022)
[22] Scells, H., Zhuang, S., Zuccon, G.: Reduce, Reuse, Recycle: Green Information Retrieval Research. In: Amigó, E., Castells, P., Gonzalo, J., Carterette, B., Culpepper, J.S., Kazai, G. (eds.) SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022, pp. 2825–2837, ACM (2022)
[23] Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., Lample, G.: LLaMA: Open and Efficient Foundation Language Models. CoRR **abs/2302.13971** (2023)

[24] van Hulst, J.M., Hasibi, F., Dercksen, K., Balog, K., de Vries, A.P.: REL: An Entity Linker Standing on the Shoulders of Giants. In: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 2197–2200, ACM, Virtual Event China (Jul 2020), ISBN 978-1-4503-8016-4

[25] Wiegmann, M., Wolska, M., Schröder, C., Borchardt, O., Stein, B., Potthast, M.: Trigger Warning Assignment as a Multi-Label Document Classification Problem. In: Rogers, A., Boyd-Graber, J., Okazaki, N. (eds.) Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 12113–12134, Association for Computational Linguistics, Toronto, Canada (Jul 2023)