Getting Your Indices in a Row: Full-Text Search for LLM Training Data for Real World

Inés Altemir Mariñas
IC, EPFL
Lausanne, Switzerland
<first.last>@epfl.edu

Anastasiia Kucherenko
IEM, HES-SO Valais-Wallis
Sierre, Switzerland
<first.last>@hevs.ch

Alexander Sternfeld
IEM, HES-SO Valais-Wallis
Sierre, Switzerland
<first.last>@hevs.ch

Andrei Kucharavy
II, HES-SO Valais-Wallis
Sierre, Switzerland
<first.last>@hevs.ch

Abstract

The performance of Large Language Models (LLMs) is determined by their training data. Despite the proliferation of open-weight LLMs, access to LLM training data has remained limited. Even for fully open LLMs, the scale of the data makes it all but inscrutable to the general scientific community, despite potentially containing critical data scraped from the internet.

In this paper, we present the full-text indexing pipeline for the Apertus LLM training data. Leveraging Elasticsearch parallel indices and the Alps infrastructure—a state-of-the-art, highly energy-efficient arm64 supercluster —we were able to index 8.6 T tokens out of 15.2 T used to train the Apertus LLM family, creating both a critical LLM safety tool and effectively an offline, curated, open web search engine. Our contribution is threefold. First, we demonstrate that Elasticsearch can be successfully ported onto next-generation arm64-based infrastructure. Second, we demonstrate that full-text indexing at the scale of modern LLM training datasets and the entire open web is feasible and accessible. Finally, we demonstrate that such indices can be used to ensure previously inaccessible jailbreak-agnostic LLM safety.

We hope that our findings will be useful to other teams attempting large-scale data indexing and facilitate the general transition towards greener computation.

CCS Concepts

• **Information systems** → **Data mining**; **Information retrieval**; *Search engine architectures and scalability*; *Data extraction and integration*; Search results deduplication; *Web searching and information discovery*; • **Computing methodologies** → Machine learning; • **Applied computing** → **Document searching**.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
WWW '26, Dubai, UAE
© 2026 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-XXXX-X/2026/06
https://doi.org/XXXXXXX.XXXXXXX

Keywords

Full-Text Search, Big Data, High Performance Computing, Large Language Models

ACM Reference Format:
Inés Altemir Mariñas, Anastasiia Kucherenko, Alexander Sternfeld, and Andrei Kucharavy. 2026. Getting Your Indices in a Row: Full-Text Search for LLM Training Data for Real World. In *Proceedings of Industry Tracks - The Web Conference 2026 (WWW '26).* ACM, New York, NY, USA, 11 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Modern Large Language Models (LLMs) are becoming increasingly integrated in products and services. Based on a simple principle - pretraining for next or missing tokens prediction [45, 48] - followed by universal instruction-following fine-tuning [40, 57], their impressive multidomain capabilities stem primarily from pretraining on massive web-scale datasets containing elements of texts required for instruction-following [20]. While most LLM training datasets are closed, several ubiquitous large-scale public datasets emerged, notably *Common Crawl*, and its derivatives, such as *The Pile* [16],*C4* [49], *FineWeb* [43], or *FineWeb2* [44], among many others. Common Crawl and its derivatives contribute to the training of most widely used LLMs, such as GPT-3, for which it supplied 80% of the data [6].

However, the training data scale is a double-edged sword. The same scale that enables the emerging LLM multidomain capabilities also enables highly problematic emerging behaviors [2]. The web-mined corpora, often not publicly shared, inevitably contain problematic content leading to problematic LLM behaviors [27, 33], that must be carefully analyzed to understand model capabilities and risks [17, 35].

Beyond the unprompted LLM toxicity arising from the inclusion of the baseline social media toxicity into the crawls [18, 32], training data has been discovered to contain private, copyrighted, abusive, or intentionally misleading information, among others. Personal information, such as names, social security numbers, and contacts, leaking from training data has been successfully extracted from GPT-2 and ChatGPT [9, 37], following their extensive memorization abilities from single examples [7, 8]. New York Times has blocked OpenAI's crawler and sued them over unauthorized copyrighted content use for model training [47]. Child Sexual Abuse Material

(CSAM) has been discovered in the training data of several models [10]. Malicious actors have been discovered to inject disinformation into LLM training and reference data, such as Kremlin-affiliated Pravda Network [50].

Problematic training data leads to problematic LLM behaviors. Even for API-only models with generation-time filtering and guardrails, jailbreaking and imperfect guardrails have led to highly problematic content generation, resulting in disastrous real-world impacts [41]. For open-weight models, such protection is impossible, and not only can they be easily jailbroken [31], especially in a multilingual setting [56], but any alignment tuning can be removed through post-training [53], unlocking problematic capabilities acquired from the training data. As such, the only reliable approach to LLM security and safety is in-depth analysis and curation of LLM training data [39]. Yet, in practice, even for open-weight models, systematic examination of training datasets remains extremely challenging. Widely used models typically disclose only broad information about their data sources, limiting transparency, reproducibility, and ultimately our ability to assess or guarantee their safety.

As part of the Apertus LLM [21] transparency effort, we address this issue a single LLM family at a time, by indexing for full-text search the already open and reproducible Apertus LLM training data. To achieve it, we scale up a prior approach based on Elasticsearch full-text indexing allowing for fuzzy matching and logic operations [14], leveraging the Alps supercluster Machine Learning Platform, built and managed by the Swiss National Supercomputing Centre (CSCS) [30, 51].

Our contributions are threefold:

- *Elasticsearch on ARM64:* We provide a technical overview of challenges, solutions, and configuration overviews for deploying Elasticsearch on ALPS, an ARM64-based GH200 high-performance computer (HPC) supercluster. While ARM64 offers future-ready energy efficiency, it poses compatibility challenges even for the most common software developed and deployed on x86/amd64 systems.
- *Indexing and performance evaluation:* We index an 8.6 T token dataset of the 15.2 T Apertus training and pretraining dataset, doubling the size compared to the previous largest full-text index of that type, Infinigram [24], and quadrupling the previous Elasticsearch-based one, WhatIsInMyBigData [14]. We open-source our code [1] and report the performance during indexing and search for a range of deployment configurations, allowing for replication and future work on web-scale data indexing.
- *Security and safety use cases:* We demonstrate two LLM safety and security usecases of LLM full-text indices, focusing on derogotary language in a multilingual setting and helpfulness for indiscriminate weapons creation.

By shifting attention from post-training alignment to the foundational quality of training data, our approach provides the scalable infrastructure needed for safer, more transparent, and more accountable AI development.

[1] https://github.com/Reliable-Information-Lab-HEVS/apertus-pretraining-data-indexing

2 Related Work

The main technical challenge in creating pipelines for transparent auditing of LLM training data is the massive size of these datasets. As a result, many existing tools attempt to overcome these limitations in various ways. Some methods perform approximate membership inference to reduce computational costs, such as Data Portraits [29]. Others analyze only small statistical samples—for example, 1% of Common Crawl (circa 81 GB), such as Luccioni and Viviano in [27], or a one-million-random-web-pages sample in [33]. While such approaches are excellent for exploratory safety work, guaranteeing LLM safety and security requires full training dataset coverage.

Several studies approached smaller datasets. Tools such as Data Measurements on HuggingFace [28] and Know Your Data from Google [19] were created to automate and improve data documentation for smaller, specific datasets. Additionally, in-depth analyses have been performed on specialized datasets, such as the COVID-19 Open Research Dataset [59], the QuoteBank corpus [55], and medical data [38]. However, none of them have been developed for the scale the modern SotA LLM training requires.

World-Wide-Web scale data indexing for LLM safety and security has gained increasing interest in recent years. For example, Dodge et al. in [12] analyzed multiple problematic aspects of the C4 corpus(156 B tokens, English-only) and created an Elasticsearch full-text index for it. Piktus et al. in [46] were the first to offer both fuzzy and exact search on ROOTS, a 1.6 TB multilingual text corpus. The WhatIsInMyBigData tool [14] applied ElasticSearch to several datasets, including C4 (156 B Tokens), RedPajama (1.4 T tokens), and The Pile (380 B tokens). While laudable initiatives, covering full training data of older LLMs, such as Eleuther GPT, BLOOM, or Pythia families [3, 4, 36], these approaches have not scaled to the SotA multi-trillion token territory.

Finally, most recently, Infinigram [24] has used Burrow-Wheeler Transform-like suffix arrays, enabling millisecond-fast exact search on datasets such as OLMo 2 Instruct (4.6 T tokens), Dolma-v1.7 (2.6 T tokens), RedPajama, and The Pile. While an impressive performance and arguably the only SotA LLM training dataset index, Infinigram only allows for exact text matching. In turn, this makes fuzzy search and logic operators significantly more involved, which is problematic given that numerous downstream tasks require them, such as training data attribution of LLM-generated texts [58].

3 Data and Tools

We first describe the datasets that were indexed and the datasets of search queries used for analysis (Section 3.1), then present Elasticsearch, the full-text search system employed for indexing and search (Section 3.2), and finally outline the technical details of the computing environment used in this work (Section 3.3). All the code used to perform the indexing and querying of the database is available https://github.com/Reliable-Information-Lab-HEVS/apertus-pretraining-data-indexing.

3.1 Data

3.1.1 FineWeb. The FineWeb dataset [42] is derived from Common Crawl, and serves as the foundation for large-scale multilingual web text pretraining. It aggregates web documents from a diverse set

of sources, applying filtering and deduplication to ensure linguistic diversity and textual quality. In the training of Apertus, three additional datasets derived from FineWeb are used: FineWeb-Edu, FineWeb-2 and FineWeb-2-HQ [26, 34, 44]. FineWeb-Edu filters educational web pages from the FineWeb dataset. Additionally, an educational quality classifier based on `Llama3-70B-Instruct` [13] is used to score each page, for further filtering. For Apertus training only pages with an educational quality score of 2 or higher are considered. Last, FineWeb-2-HQ considers the top 10% quality documents of FineWeb-2 for the top 20 highest resource languages, based on a classifier using XLM-RoBERTa embeddings, and re-hydrades the deduplicated dataset, adding up to 8 copies of abundant high-quality documents [34].

3.1.2 Apertus training data. Apertus pretraining data underwent additional processing before being used in training. First, copyrighted material remaining in the training datasets is identified and removed. Second, data from from websites that have opted out of crawling by AI-related crawlers as of January 2025 is removed as well. Third, all detected email addresses, IP addresses and IBAN bank account numbers are replaced with anonymous markers. Finally, toxicity filtering is performed across nine languages, with custom-trained language-specific classifiers, with data for low-resource languages being preserved as is. The full pipeline for the creation of the training data has been made publicly available [2] and more information on each of these filtering steps is described in the technical report of Apertus [21].

3.1.3 Indexed datasets. In this section, we describe the used subset of the Apertus training data. In total, we indexed a text volume of 8.6 T tokens, corresponding to 58% of the current total training data (15 T), providing a dataset comparable in scale to the full training corpus. This enables the characterization of indexing and analysis efficiency on large-scale data.

Specifically, we indexed all datasets included in Phase 1 of the Apertus pretraining, as described in the technical report [21]. While Phase 1 utilized approximately 5 T tokens, the datasets we indexed correspond to supersets that were also sampled in later phases, resulting in a higher Apertus training data coverage on our side. Table 1 displays the different components. The most prominent datasets are FineWeb-Edu [26], the 33% documents of highest quality from FineWeb-2-HQ [34], and for the languages not included in FineWeb-2-HQ a randomly sampled 33% from FineWeb-2 [44]. Additionally, StarCoder code-specific dataset was included [23], along with a Common Crawl math-focused subset, extracted as part of SmolLM2 LLM training [1]. Finally, the permissively licensed books from Project Gutenberg [15] data is added to the pretraining mix to evaluate the memorization of book-like content, while Poison data is performance-neutral data added as part of persistent pretraining data poisoning research [60].

[2] https://github.com/swiss-ai/pretrain-data

Dataset	Tokens (B)
FineWeb-Edu (Score-2)	4815
FineWeb-2-HQ (33% highest quality) and FineWeb-2 (33% sample of other languages)	3557
StarCoder	235
FineMath CommonCrawl subset	32
Gutenberg and Poison	2

Table 1: Overview of the datasets indexed, along with the total number of tokens in each dataset.

The core strength of both Apertus and FineWeb-2 dataset is the attention to multilinguality. For this reason, we also focus our attention in Section 6 on the presence of harmful content across various languages. Given that Apertus was developed in Switzerland, in addition to English we consider Swiss German (`gsw`) and the official Swiss languages (Italian (`ita`), German (`deu`), French (`fra`)) that are high-resource. Additionally, we attempt to sample non-european and lower-resource languages by focusing on Arabic, Thai, Kabyle, and Filipino. Finally, we investigate Esperanto as an example of synthetic language without a native speaker population.

3.1.4 Datasets of search queries. One goal of the indexing and search pipeline is to examine the presence and distribution of potentially harmful or otherwise problematic content within the Apertus training data. For this, we focus on three LLM security and safety-related use cases:

- **Weaponized Words dictionary**[3]: Is a large multilingual lexicon of curse words, slurs, and other forms of toxic or discriminatory language, maintained covering over 137 languages across 236 countries. Table 2 shows the number of terms for each of the languages we consider: English, Italian, French and German. The lengths of the harmful terms range from 1 to 3 words. Note that we cannot publicly disclose this dataset, as access is restricted to research teams upon special request per the Weaponized Word Terms of Service.
- **Obscene Words list [52]**: The *List of Dirty, Naughty, Obscene, and Otherwise Bad Words* (LDNOOBW) is a collection of profane and offensive terms used for cleaning of the Colossal Clean Crawled Corpus [11]. The strength of the dataset is that it contains contains terms for 28 languages, including several low-resource languages such as Kabyle and Thai. The length of the profanities ranges from 1 to 5 words. In our analysis we consider the languages English, Italian, French, German, Arabic, Filipino, Esperanto, Kabyle and Thai. Table 2 shows for each language the number of profanities that is included in the dataset.
- **Chemical weapons dataset**: A curated list of dangerous chemical agents selected from *A Laboratory History of Chemical Warfare Agents* by Jared Ledgard [22]. Containing detailed synthesis instruction of chemical warfare agents, this book has only been freely available on the internet and only recently was de-indexed, with excerpts from it likely being included into LLM pretraining dataset, potentially making

[3] https://weaponizedword.org/

them helpful for indiscriminate weapons creation. We focus on several simple blood agents and their precursors to find potentially problematic contents in the internet, while leaving general-purpose chemistry knowledge intact. The dataset displayed in Table 3 contains 17 terms, and we focus on several of the languages most likely to be used within Switzerland to look for this type of information, notably Dutch, Spanish, Serbo-Croatian, and Portuguese.

For each dataset, we report raw occurrence counts rather than terms prevalence, given that prior research suggests that knowledge acquisition and memorization are determined by information occurrence rather than prevalence [9].

	Number of terms	
Language	LDNOOBW	Weaponized Words
English	403	592
Italian	168	21
French	91	44
German	66	31
Arabic	38	-
Esperanto	37	-
Thai	31	-
Kabyle	22	-
Filipino	14	-

Table 2: Overview of the languages from the LDNOOBW and Weaponized Words datasets, showing the number of profanities for each language.

Chemical term
Chloropicrin
Bromopicrin
Diphenylchloroarsine
Adamsite
Hydrogen cyanide
Cyanogen chloride
Phosgene
Sulfur mustard
Methyldichloroarsine
Acetone peroxide
Triacetone triperoxide
Flash powder
Potassium Perchlorate
Potassium Chlorate
Glycerine
Nitric Acid
Cyanogen Bromide

Table 3: Manually curated list of dangerous chemical agents from *A Laboratory History of Chemical Warfare Agents* [22]

3.2 Elasticsearch

3.2.1 General idea. Elasticsearch is a distributed search and analytics engine designed for efficiently handling large-scale text and structured data, commonly considered as a NoSQL database or a local search engine. At its core, it transforms raw information into an optimized index that allows users to perform fast and flexible queries, ranging from simple keyword lookups to complex semantic searches. This indexing-and-search model makes it especially well-suited for working with massive, heterogeneous datasets, such as collections of web-scraped documents, leading to its choice by several previous projects analyzing LLM training data.

In the described pipeline, raw data is stored as Parquet files and streamed into the indexing process to avoid overwhelming memory resources. Before being stored, the text undergoes multiple levels of processing via configurable analyzers, which generate different searchable representations — from normalized text suitable for semantic search, to exact forms that preserve original formatting and structure.

Scalability is achieved through Elasticsearch's distributed architecture. Data is partitioned into shards and indexed in parallel, allowing multiple workers to process separate portions of the dataset simultaneously. Performance is further enhanced through techniques such as bulk indexing, adjustable batch sizes, and dynamic refresh intervals, which balance throughput and responsiveness during large-scale operations.

For datasets too large to fit on a single node, advanced cluster management strategies allow a scalable search cluster. Separate Elasticsearch instances can each index a subset of the data, and later be merged into a unified search space using remote `_reindex` operations. This approach maintains data integrity while allowing the system to scale beyond the limitations of individual nodes.

3.2.2 Parameter Tuning. The indexing operation relies on the `elasticsearch.helpers.parallel_bulk` function, which distributes bulk requests across multiple threads to enable concurrent ingestion into Elasticsearch. Achieving good runtime and memory performance requires careful tuning of several interdependent parameters, as poor settings can easily create CPU under-utilization, memory exhaustion, or request bottlenecks.

Thread count determines the number of worker threads handling bulk requests. Each thread maintains its own request queue, which increases memory consumption proportionally. The value should not exceed the number of available CPU cores. Higher thread counts improve concurrency but also risk memory pressure if chunk sizes are large.

Chunk size specifies the number of documents in each bulk request. Small chunks create more frequent bulk submissions, increasing overhead from request construction and processing inside Elasticsearch. Oversized chunks, on the other hand, increase memory usage and the chance of timeouts. The maximum feasible chunk size is bounded by the ratio of `max_chunk_size` to `avg_doc_size`:

$$chunk\_size \leq \frac{max\_chunk\_size}{avg\_doc\_size} \tag{1}$$

Max chunk bytes controls the maximum payload size of a bulk request. Larger values reduce the relative cost of bulk request

management inside Elasticsearch but require more RAM per thread to hold the request in memory.

Queue size defines the buffer between the main thread (producing chunks) and worker threads (processing them). A larger queue helps absorb temporary imbalances between production and consumption, but also increases memory footprint. In practice, values between 2 and 8 were tested.

Parameter choices must therefore balance CPU parallelism, request management overhead, and memory constraints. Estimating memory cost from `avg_doc_size` and bounding chunk size via `max_chunk_size` provides a practical starting point, after which empirical tuning is essential to reach stable throughput without exceeding hardware limits. **Hardware considerations** must also be taken in account, potentially providing an upper limit on document insertion speed. While Elasticsearch uses a two-phase atomic commit protocol for single document insertion it does not natively support bulk atomic transactions[4], meaning that in a multi-threaded environment, each document insertion required two sequential round-trips to the storage for each inserted document. On well-optimized storage infrastructure such as ALPS IOPStore we used to indexing, this drive access latency can approach 50 microseconds, with expected maximal indexing speeds around 10 000 documents per second. This limit is determined by hardware latencies rather than document size or indexing settings.

3.2.3 Types of Queries. With Elasticsearch, one has the possibility to perform various types of queries. For instance, one can perform exact queries, fuzzy queries, or perform boolean logic with manually configured boolean queries. In this work, we focus on the match phrase query, which is described in more detail below. For this query, the system processes content using the `web_content_analyzer`, which performs:

- HTML stripping to remove tags
- Standard tokenization to split text into words
- Lowercase normalization
- ASCII folding to convert accented characters to plain letters

The `match_phrase_query` searches for an exact sequence of words, with a configurable tolerance for how closely those words must appear together. This tolerance is controlled by the `MATCH_PHRASE_SLOP` parameter. For single-word queries, it behaves the same as a regular match query. For multi-word phrases such as "climate change," the terms must appear in the same order after analysis. When the slop value is set to 0, the words must occur directly next to each other. For example, "climat" must immediately precede "chang." Increasing the slop allows a greater distance between words: a slop of 1 permits one intervening word, and a slop of 2 allows two. In practice, this means a slop of 1 can match "climate and change," while a slop of 2 can match "climate action and change."

3.3 Computing Environment and Resource Usage

3.3.1 Computing environment. The experiments were conducted on the Alps Research Infrastructure at the Swiss National Supercomputing Centre (CSCS). The large-scale high-performance computing (HPC) system is an HPE Cray EX system with a HPL performance of 434 PFlops, making it one of the most powerful existing supercomputers, sitting as the N 8 of Top 500 [5].

The system is equipped with Arm64-based NVIDIA Grace Hopper GH200 nodes. Each node combines Grace CPUs with integrated Hopper GPUs, providing four Grace-Hopper modules and associated network interfaces. In total, the system contains 10,752 nodes, with each node containing 4 GPUs and 4 CPU sockets. The storage includes a 100PB ClusterStor HDD system and a 3PB ClusterStor SSD system, alongside a 1PB VAST storage system. Sharing the architecture with the current top 1 system on the Green Top 500 list, Alps itself is highly energy-efficient, achieving 61 GFlops/watt and ranking 19th on that list[6].

Architecturally, Alps uses a software-defined cluster (vCluster), abstracting infrastructure, service management and user environments, bridging the gap between traditional HPCs and Cloud service provider deployments. Specifically, the ML-dedicated vCluster, Clariden, is a container-first cluster using Slurm as scheduler [51] Elasticsearch version 7.17.28 was deployed in a distributed setup across multiple nodes.

3.3.2 Usage and emissions. We give an estimation of the CO_2 emitted by the computation. In total, we used 7741 node hours for indexing and search presented here, including trial runs and parameter tuning, totaling less than 0.1% of the computational power used to train the Apertus Model. We assume a power usage of 560W per node, consistent with Apertus training load, resulting in 4.3 MWh for the development and execution of the data indexing. While Alps' electric supply is carbon-neutral [54], by performing a consumption substitution analysis, we estimate that CO_2 emissions per kilowatt-hour (kWh) in Switzerland are on average 21g CO_2eq/kWh, our work led to approximately 90kg CO_2eq, or around 0.225% of the yearly emissions of an average Swiss person [5, 25].

4 Elasticsearch Deployment

Deploying Elasticsearch on high-performance computing (HPC) systems equipped with Grace Hopper nodes and Arm64 architectures presents several nontrivial challenges that do not arise in more conventional x86-based environments. A central difficulty lies in containerization. Many HPC platforms provide their own container engines, optimized for workload scheduling and security, but incompatible with Docker. This incompatibility reflects both architectural and security concerns: the majority of Docker images target x86_64/amd64 rather than Arm64, and Docker's shared read/write permissions are unsuitable for multi-user HPC systems. As a result, the official Elasticsearch image and orchestration tools such as Docker Compose cannot be used.

To overcome these limitations, we built custom OCI-compliant container images using Podman[7]. These images bundled the appropriate version of Elasticsearch together with supporting tools such as Python3, pyarrow, and curl, while remaining fully compatible with the Arm64 architecture and the HPC system's security model. Orchestration, normally simplified by Docker Compose, was

[4] https://www.elastic.co/blog/found-elasticsearch-as-nosql

[5] https://www.top500.org/lists/top500/list/2025/06/

[6] https://www.top500.org/lists/green500/list/2025/06/

[7] https://podman.io/

		Data size (GB)	Time (h)	Indexing rate (doc/s)	Index size/data size	Avg. peak memory (GB)
Fineweb-2 Edu (EN)	Score 2	12,736.9	143.7	10,296.4	1.3	4.9
Fineweb-2 Europe	High quality*	2,660.0	408.3	589.4	1.1	7.5
	Medium quality	21	3.2	1,932.5	2.2	5.5
Fineweb-2 Other	High quality	991.8	194.4	1,315.4	2.8	2.4
	Medium quality	76.6	0.8	5,868.3	2.3	8.0
Finemath-3		47	0.4	11,062	1.7	7.4
Starcoder		229.1	4.2	10,919.3	1.4	12.7
Gutenberg		3.2	0.05	1,302.2	2.4	4.9
Poison		0.6	0.05	2,410.3	1.6	2.4

Table 4: Time required to index each part of the phase 1 training data, alongside the data size, indexing rate, overhead, and average peak memory. * Designates the deduplicated dataset index.

re-implemented through SLURM job definitions, ensuring reproducibility and integration with the resource manager.

In an HPC environment, system-level restrictions can give further challenges. In our case, environment definition files, typically used to propagate configuration variables into containers, were not reliably interpreted by Elasticsearch. We addressed this by redirecting all runtime parameters through explicit command-line arguments injected at container startup. Similarly, Elasticsearch's reliance on memory mapping conflicted with the immutable kernel parameter `vm.max_map_count`, which was set too low on the cluster to satisfy bootstrap checks. We resolved this by disabling the memory mapping, enabling Elasticsearch to start successfully. However, the disadvantage is that due to the absence of memory mapping we observed a reduction in the I/O efficiency and a higher system memory usage.

Last, networking required careful reconfiguration. In this HPC environment, HTTP traffic was subject to proxy mediation, leading to unexpected failures when attempting to connect to Elasticsearch's default endpoint on localhost:9200. We resolved this by explicitly bypassing the proxy for local traffic, binding all Elasticsearch interfaces strictly to 127.0.0.1, and disabling multi-node discovery to comply with SLURM's job isolation model. These settings ensured stable single-node operation suitable for HPC workloads.

5 Performance Statistics

We start by considering the indexing and search performance of Elastic Search on the Phase 1 training data, as described in Section 3.1.3. To this end, we perform three analyses. First, we look at the indexing statistics for each of the datasets. Then, we consider the effect of the query length on the search time.

5.1 Indexing Performance

Table 4 displays the indexing statistics for each of the datasets described in Section 3.1.3. The indexing performance varies notably with the linguistic composition and content type. Indexing pure English text proceeds substantially faster than indexing a multilingual dataset comprising a variety of European languages. The throughput for English text reached 10,297 documents per second, whereas the multilingual dataset achieved only 589 documents per second. This difference likely arises from the higher linguistic and computational complexity of multilingual data. English text is comparatively uniform in structure and encoding, with straightforward tokenization, while multilingual corpora introduce additional processing overhead due to diverse character sets, diacritics, and language-specific normalization routines.

Memory usage measurements show that the peak memory consumption is considerably higher when indexing code than when indexing natural language text. Code contains more unique tokens, longer identifiers, and complex syntactic patterns, which require larger intermediate data structures and less redundancy. As a result, code indexing demands greater memory resources.

The index overhead also differs significantly between datasets. For pure English, the ratio between index size and raw data size is relatively low (1.3), whereas it significantly increases when multiple languages are included. This indicates that English text is more easily compressed and represented efficiently, while multilingual datasets require larger vocabularies and less compressible structures due to linguistic variability.

Given that Fineweb-2-HQ was partially rehydrated for duplicated content, we verified if content de-duplication could be leveraged to compress the index size and accelerate the indexing on Fineweb-2 Europe High quality only. For this, we pre-computed full document SHA-256 hashes and inserted only the first document with the matching SHA-256 hash. Overall, we observed that approximately 68% was duplicates, with most replicated content containing up to 8 copies, consistent with Fineweb-2-HQ documentation. Rather than increase in indexing speed, we observed an indexing slow-down proportional to duplicated content prevalence (approximately 68%) lost in Fineweb-2 Europe High vs non-deduplicated Fineweb-2 Europe Medium, cf Table 4. We interpret it as Elasticsearch managing duplicates internally with minimal overhead.

5.2 Query Length and Search Time

In Figure 1 we evaluate the performance of match phrase queries when applied to an index with varying query segment length. To this end, for 13 different lengths between 1 and 300 words, we sampled 25 queries from the source datasets used to build the indexes. These segments were then issued as match phrase queries against the corresponding indexes to calculate the average search time and the standard deviation interval. We can observe a logical tendency of longer queries taking longer time, and get an intuition for overall performance.

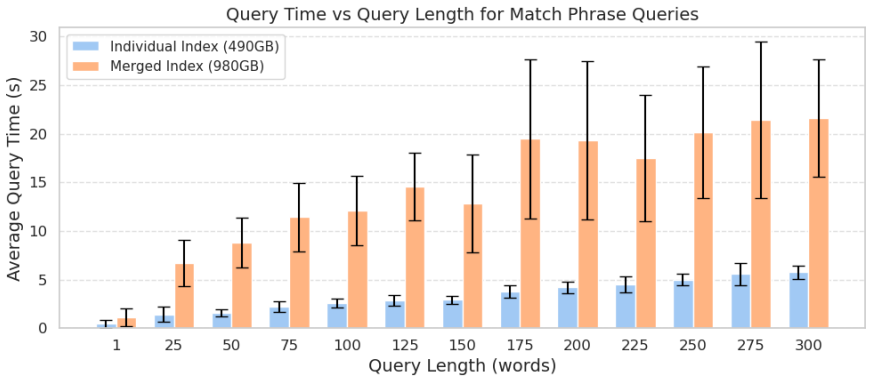

Figure 1: Query time vs query length for match phrase queries based on the index size. Averages +/- std.

6 Presence of Harmful terms in Apertus Pretraining Data

Warning: This section contains samples of offensive words outside context.

In this section, we describe the analysis of the presence of harmful terms in the phase 1 training data of Apertus. To this end, we consider three Weaponized Words (WW), the manually constructed list of chemical warfare substances, and the LDNOOBW. Table 5 shows the number of documents that contain at least one of the terms in each dataset. As expected, the results show that in low-resource languages there is fewer presence of harmful content.

Language	Count		
	WW (x1,000,000)	Chemical (x1,000,000)	LDNOOBW (x1,000,000)
English	1,245.8	2.96	661.6
Italian	1.6	0.23	18.5
French	16.8	0.12	202.5
German	9.9	0.58	14.9
Dutch	-	0.12	-
Spanish	-	0.22	-
Serbo-Croatian	-	0.04	-
Portuguese	-	0.22	-
Arabic	-	-	1.7
Filipino	-	-	0.6
Esperanto	-	-	3.1
Kabyle	-	-	0.0001
Thai	-	-	0.3

Table 5: Counts of document including terms deemed harmful according to: Weaponized Words (WW), Chemical substances and the LDNOOBW, Millions of hits.

Furthermore, we observe that despite apparent high counts of problematic content, most highly represented words are general terms that may not always be problematic. These terms encompass general, important themes and cannot be universally removed from the training data without severely degrading models' performance, which is consistent with previous reports [11]. Driving this point further, despite the abundance of toxic words, Apertus performs well on toxicity evaluations [21] (Section 5.2, Table 26). However, full-text indexing enables fine-grained thematic analyses for granular further training data filtering and LLM behavior guarantees.

6.1 Weaponized Words

Figure 2 shows, for each of the languages taken into consideration, the five most prominent terms in the training data. We observe that there are common terms across different languages, such as "dying" (*mourir, sterben*) and "killing" (*uccidere, toten*). At the same time, there is a clear difference for the English language, in which terms related to *sex* are the most common in the training data.

6.2 Chemical Substances

We observe that the counts are exceptionally high for common chemical compounds widely known to the general public, notably Glycerine, Nitric Acid, and Hydrogen Cyanide. In particular, most of their mentions would occur outside chemical weapons synthesis instructions. We observe that for rare terms unequivocally connected to chemical weapons synthesis (Bromopicrin, Methyldichloroarsine), substantial counts occur outside of English, indicating that multilingual data curation is essential to ensure model safety.

Table 2 displays the most common find terms from the LDNOOBW for each of the languages. Similar to Weaponized Words, we observe high counts of general words, which are necessary for discussing important topics.

7 Conclusion and Future Work

In this paper, we demonstrate how a full-text search index for 8.6 T multilingual tokens can be built on modern energy-efficient infrastructure and how it can be leveraged to ensure LLM behavior safety. Our contribution is threefold. First, we demonstrate that, despite initial difficulties, common server-side applications – such as Elasticsearch – can be run on next-generation architectures. We believe this case study, which ported a common business application to an energy-efficient next-generation architecture, demonstrates a high readiness for a general green computing transition, critical as computational power demands are soaring. Second, we demonstrate

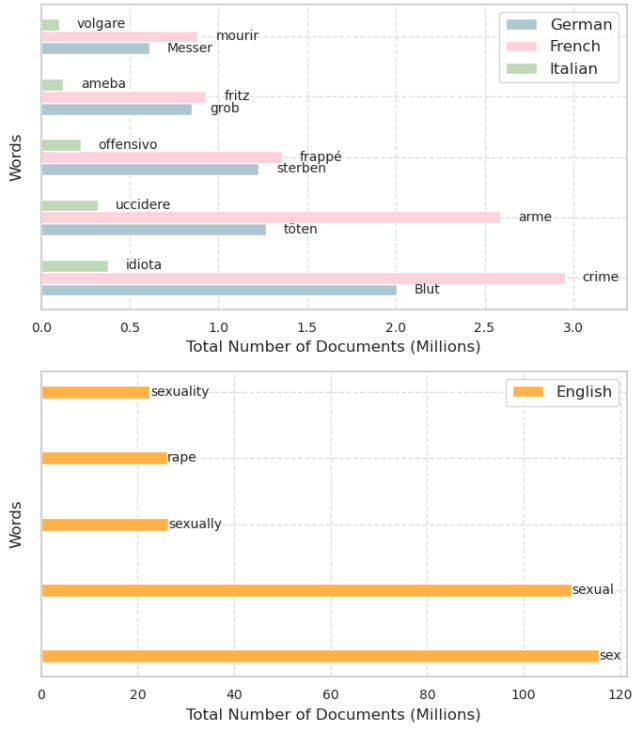

Figure 2: Top 5 Weaponized Words by total number of occurrences

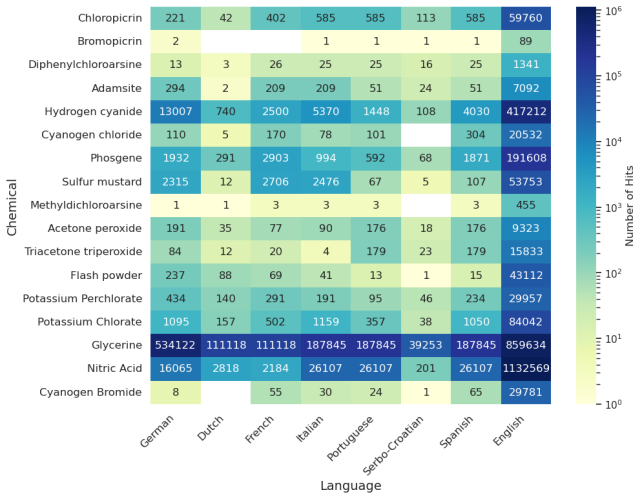

Figure 3: Heatmap displaying the presence of chemical terms in the training data for various languages.

that indexing texts at the scale of LLM training datasets is feasible for small teams using off-the-shelf software, requiring less than 0.1% of the compute used to train the LLMs in question. We hope this will encourage future analyses of the LLM training data and the inclusion of full training data analysis in LLM release standards. Finally, we demonstrate that the resulting index can be used for in-depth investigation of problematic content in the LLM training data, allowing for fine-grained context-based analysis and training data cleaning. We hope that this will lead to a common adoption of training data-based LLM safety and security mechanisms, such as those proposed by other teams, notably in [39].

Language	Top 3 terms	Total count (x 1,000,000)
English	sex, sexual, sexually	251.9
French	con, péter, bitte	189.6
German	porno, penis, fick	6.6
Filipino	bobo, tanga, burat	0.5
Kabyle	qqu, abbuc, uqan	0.0001
Thai	[Hī], [Yěd], [Khwy]	0.2
Esperanto	fik, piĉo, fek	3.0
Arabic	[faraj], [nik], [aghtisabi]	0.4

Table 6: Top 3 most found terms from the LDNOOBW for each of the languages taken into consideration, alongside the total count for those three terms. Enclosure by [] indicates transliteration.

While our training data indexing is primarily motivated by safety and security, the size of the indexed data begins to approach that of the Common Crawl dataset itself. In turn, this suggests that an offline search index for the entire open internet is within reach for smaller entities. An exciting perspective in itself, it is particularly interesting for the general-purpose factual rooting of LLM generation based on offline search curated open web subsets. In turn, this raises a host of ethical and economic questions, notably regarding the fair compensation of individuals who originally created and hosted the retrieved content, likely opening novel research directions in worldwide web economic, legal, and governance aspects.

Acknowledgements

The Swiss Supercomputing Centre (CSCS) for providing computational support and infrastructure for this project; SwissAI initiative and the Apertus team for the insight regarding the training data and computational budget allocation; and Prof. Antoine Bosselut for co-supervising IAM during the duration of the project. This work has been supported by the armasuisse S+T grant AR-CYD-C-025 to AK, AK, and AS, and a SwissAI Initiative Large Project Grant 45.

References

[1] Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlícek, Agustín Piqueres Lajarín, Vaibhav Srivastav, Joshua Lochner, Caleb Fahlgren, Xuan-Son Nguyen, Clémentine Fourrier, Ben Burtenshaw, Hugo Larcher, Haojun Zhao, Cyril Zakka, Mathieu Morlon, Colin Raffel, Leandro von Werra, and Thomas Wolf. 2025. SmolLM2: When Smol Goes Big - Data-Centric Training of a Small Language Model. *CoRR* abs/2502.02737 (2025). arXiv:2502.02737 doi:10.48550/ARXIV.2502.02737

[2] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, Jamie Kerr, Jared Mueller, Jeffrey Ladish, Joshua Landau, Kamal Ndousse, Kamile Lukosiute, Liane Lovitt, Michael Sellitto, Nelson Elhage, Nicholas Schiefer, Noemí Mercado, Nova DasSarma, Robert Lasenby, Robin Larson, Sam Ringer, Scott Johnston, Shauna Kravec, Sheer El Showk, Stanislav Fort, Tamera Lanham, Timothy Telleen-Lawton, Tom

Conerly, Tom Henighan, Tristan Hume, Samuel R. Bowman, Zac Hatfield-Dodds, Ben Mann, Dario Amodei, Nicholas Joseph, Sam McCandlish, Tom Brown, and Jared Kaplan. 2022. Constitutional AI: Harmlessness from AI Feedback. *CoRR* abs/2212.08073 (2022). arXiv:2212.08073 doi:10.48550/arXiv.2212.08073

[3] Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A Suite for Analyzing Large Language Models Across Training and Scaling. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA (Proceedings of Machine Learning Research, Vol. 202)*, Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (Eds.). PMLR, 2397–2430. https://proceedings.mlr.press/v202/biderman23a.html

[4] Sidney Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, Usvsn Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. GPT-NeoX-20B: An Open-Source Autoregressive Language Model. In *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*. Association for Computational Linguistics, virtual+Dublin, 95–136. doi:10.18653/v1/2022.bigscience-1.9

[5] Simon Bradley. 2024. How green are the Swiss? https://www.swissinfo.ch/eng/sci-&-tech/how-green-are-the-swiss/49130784

[6] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. arXiv:2005.14165 [cs.CL] https://arxiv.org/abs/2005.14165

[7] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (Vancouver, BC, Canada) *(NIPS '20)*. Curran Associates Inc., Red Hook, NY, USA, Article 159, 25 pages.

[8] Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramèr, and Chiyuan Zhang. 2022. Quantifying Memorization Across Neural Language Models. *ArXiv* abs/2202.07646 (2022). https://api.semanticscholar.org/CorpusID:246863735

[9] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, Alina Oprea, and Colin Raffel. 2021. Extracting Training Data from Large Language Models. arXiv:2012.07805 [cs.CR] https://arxiv.org/abs/2012.07805

[10] Caoilte Ó Ciardha, John Buckley, and Rebecca S. Portnoff. 2025. AI Generated Child Sexual Abuse Material – What's the Harm? arXiv:2510.02978 [cs.CY] https://arxiv.org/abs/2510.02978

[11] Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. 2021. Documenting Large Webtext Corpora: A Case Study on the Colossal Clean Crawled Corpus. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (Eds.). Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 1286–1305. doi:10.18653/v1/2021.emnlp-main.98

[12] Jesse Dodge, Maarten Sap, Ana Marasović, William Agnew, Gabriel Ilharco, Dirk Groeneveld, Margaret Mitchell, and Matt Gardner. 2021. Documenting Large Webtext Corpora: A Case Study on the Colossal Clean Crawled Corpus. arXiv:2104.08758 [cs.CL] https://arxiv.org/abs/2104.08758

[13] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. 2024. The Llama 3 Herd of Models. *CoRR* abs/2407.21783 (2024). arXiv:2407.21783 doi:10.48550/ARXIV.2407.21783

[14] Yanai Elazar, Akshita Bhagia, Ian Magnusson, Abhilasha Ravichander, Dustin Schwenk, Alane Suhr, Pete Walsh, Dirk Groeneveld, Luca Soldaini, Sameer Singh, Hanna Hajishirzi, Noah A. Smith, and Jesse Dodge. 2024. What's In My Big Data? arXiv:2310.20707 [cs.CL] https://arxiv.org/abs/2310.20707

[15] Manuel Faysse. 2023. Project Gutenberg dataset. https://huggingface.co/datasets/manu/project_gutenberg

[16] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. 2020. The Pile: An 800GB Dataset of Diverse Text for Language Modeling. arXiv:2101.00027 [cs.CL] https://arxiv.org/abs/2101.00027

[17] Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. 2021. Datasheets for datasets. *Commun. ACM* 64, 12 (Nov. 2021), 86–92. doi:10.1145/3458723

[18] Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A. Smith. 2020. RealToxicityPrompts: Evaluating Neural Toxic Degeneration in Language Models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association for Computational Linguistics, Online, 3356–3369. doi:10.18653/v1/2020.findings-emnlp.301

[19] Google. 2021. Know Your Data. https://github.com/pair-code/knowyourdata. GitHub repository.

[20] Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. 2023. Textbooks Are All You Need. *CoRR* abs/2306.11644 (2023). arXiv:2306.11644 doi:10.48550/ARXIV.2306.11644

[21] Alejandro Hernández-Cano, Alexander Hägele, Allen Hao Huang, Angelika Romanou, Antoni-Joan Solergibert, Barna Pasztor, Bettina Messmer, Dhia Garbaya, Eduard Frank Ďurech, Ido Hakimi, Juan García Giraldo, Mete Ismayilzada, Negar Foroutan, Skander Moalla, Tiancheng Chen, Vinko Sabolčec, Yixuan Xu, Michael Aerni, Badr AlKhamissi, Ines Altemir Marinas, Mohammad Hossein Amani, Matin Ansaripour, Ilia Badanin, Harold Benoit, Emanuela Boros, Nicholas Browning, Fabian Bösch, Maximilian Böther, Niklas Canova, Camille Challier, Clement Charmillot, Jonathan Coles, Jan Deriu, Arnout Devos, Lukas Drescher, Daniil Dzenhaliou, Maud Ehrmann, Dongyang Fan, Simin Fan, Silin Gao, Miguel Gila, María Grandury, Diba Hashemi, Alexander Hoyle, Jiaming Jiang, Mark Klein, Andrei Kucharavy, Anastasiia Kucherenko, Frederike Lübeck, Roman Machacek, Theofilos Manitaras, Andreas Marfurt, Kyle Matoba, Simon Matrenok, Henrique Mendoncça, Fawzi Roberto Mohamed, Syrielle Montariol, Luca Mouchel, Sven Najem-Meyer, Jingwei Ni, Gennaro Oliva, Matteo Pagliardini, Elia Palme, Andrei Panferov, Léo Paoletti, Marco Passerini, Ivan Pavlov, Auguste Poiroux, Kaustubh Ponkshe, Nathan Ranchin, Javi Rando, Mathieu Sauser, Jakhongir Saydaliev, Muhammad Ali Sayfiddinov, Marian Schneider, Stefano Schuppli, Marco Scialanga, Andrei Semenov, Kumar Shridhar, Raghav Singhal, Anna Sotnikova, Alexander Sternfeld, Ayush Kumar Tarun, Paul Teiletche, Jannis Vamvas, Xiaozhe Yao, Hao Zhao Alexander Ilic, Ana Klimovic, Andreas Krause, Caglar Gulcehre, David Rosenthal, Elliott Ash, Florian Tramèr, Joost VandeVondele, Livio Veraldi, Martin Rajman, Thomas Schulthess, Torsten Hoefler, Antoine Bosselut, Martin Jaggi, and Imanol Schlag. 2025. Apertus: Democratizing Open and Compliant LLMs for Global Language Environments. arXiv:2509.14233 [cs.CL] https://arxiv.org/abs/2509.14233

[22] J.B. Ledgard. 2006. *The Laboratory History of Chemical Warfare Agents*. Paranoid Publications Group.

[23] Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, Qian Liu, Evgenii Zheltonozhskii, Terry Yue Zhuo, Thomas Wang, Olivier Dehaene, Mishig Davaadorj, Joel Lamy-Poirier, João Monteiro, Oleh Shliazhko, Nicolas Gontier, Nicholas Meade, Armel Zebaze, Ming-Ho Yee, Logesh Kumar Umapathi, Jian Zhu, Benjamin Lipkin, Muhtasham Oblokulov, Zhiruo Wang, Rudra Murthy V, Jason T. Stillerman, Siva Sankalp Patel, Dmitry Abulkhanov, Marco Zocca, Manan Dey, Zhihan Zhang, Nour Fahmy, Urvashi Bhattacharyya, Wenhao Yu, Swayam Singh, Sasha Luccioni, Paulo Villegas, Maxim Kunakov, Fedor Zhdanov, Manuel Romero, Tony Lee, Nadav Timor, Jennifer Ding, Claire Schlesinger, Hailey Schoelkopf, Jan Ebert, Tri Dao, Mayank Mishra, Alex Gu, Jennifer Robinson, Carolyn Jane Anderson, Brendan Dolan-Gavitt, Danish Contractor, Siva Reddy, Daniel Fried, Dzmitry Bahdanau, Yacine Jernite, Carlos Muñoz Ferrandis, Sean Hughes, Thomas Wolf, Arjun Guha, Leandro von Werra, and Harm de Vries. 2023. StarCoder: may the source be with you! *Trans. Mach. Learn. Res.* 2023 (2023). https://openreview.net/forum?id=KoFOg41haE

[24] Jiacheng Liu, Sewon Min, Luke Zettlemoyer, Yejin Choi, and Hannaneh Hajishirzi. 2025. Infini-gram: Scaling Unbounded n-gram Language Models to a Trillion Tokens. arXiv:2401.17377 [cs.CL] https://arxiv.org/abs/2401.17377

[25] LowCarbonPower. 2025. Electricity in Switzerland in 2024/2025. https://lowcarbonpower.org/region/Switzerland

[26] Anton Lozhkov, Loubna Ben Allal, Leandro von Werra, and Thomas Wolf. 2024. *FineWeb-Edu*. https://huggingface.co/datasets/HuggingFaceFW/fineweb-edu

[27] Alexandra Luccioni and Joseph Viviano. 2021. What's in the Box? An Analysis of Undesirable Content in the Common Crawl Corpus. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli (Eds.). Association for Computational Linguistics, Online, 182–189. doi:10.18653/v1/2021.acl-short.24

[28] Sasha Luccioni, Yacine Jernite, and Margaret Mitchell. 2021. Data Measurements Tool. https://huggingface.co/blog/data-measurements-tool. Hugging Face Blog.

[29] Marc Marone and Benjamin Van Durme. 2023. Data Portraits: Recording Foundation Model Training Data. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*. https://arxiv.org/abs/2303.03919

[30] Maxime Martinasso, Mark Klein, and Thomas C. Schulthess. 2025. Alps, a versatile research infrastructure. *CoRR* abs/2507.02404 (2025). arXiv:2507.02404 doi:10.48550/ARXIV.2507.02404

[31] Mantas Mazeika, Long Phan, Xuwang Yin, Andy Zou, Zifan Wang, Norman Mu, Elham Sakhaee, Nathaniel Li, Steven Basart, Bo Li, David Forsyth, and Dan Hendrycks. 2024. HarmBench: A Standardized Evaluation Framework for Automated Red Teaming and Robust Refusal. arXiv:2402.04249 [cs.LG] https://arxiv.org/abs/2402.04249

[32] Kris McGuffie and Alex Newhouse. 2020. The Radicalization Risks of GPT-3 and Advanced Neural Language Models. arXiv:2009.06807 [cs.CY] https://arxiv.org/abs/2009.06807

[33] Sai Krishna Mendu, Harish Yenala, Aditi Gulati, Shanu Kumar, and Parag Agrawal. 2025. Towards Safer Pretraining: Analyzing and Filtering Harmful Content in Webscale datasets for Responsible LLMs. arXiv:2505.02009 [cs.CL] https://arxiv.org/abs/2505.02009

[34] Bettina Messmer, Vinko Sabolčec, and Martin Jaggi. 2025. Enhancing Multilingual LLM Pretraining with Model-Based Data Selection. *arXiv* (2025). https://arxiv.org/abs/2502.10361

[35] Margaret Mitchell, Simone Wu, Andrew Zaldivar, Parker Barnes, Lucy Vasserman, Ben Hutchinson, Elena Spitzer, Inioluwa Deborah Raji, and Timnit Gebru. 2019. Model Cards for Model Reporting. In *Proceedings of the Conference on Fairness, Accountability, and Transparency* (Atlanta, GA, USA) *(FAT* '19)*. Association for Computing Machinery, New York, NY, USA, 220–229. doi:10.1145/3287560.3287596

[36] Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, Xiangru Tang, Dragomir Radev, Alham Fikri Aji, Khalid Almubarak, Samuel Albanie, Zaid Alyafeai, Albert Webson, Edward Raff, and Colin Raffel. 2022. Crosslingual Generalization through Multitask Finetuning. arXiv:2211.01786 https://arxiv.org/abs/2211.01786

[37] Milad Nasr, Javier Rando, Nicholas Carlini, Jonathan Hayase, Matthew Jagielski, A. Feder Cooper, Daphne Ippolito, Christopher A. Choquette-Choo, Florian Tramèr, and Katherine Lee. 2025. Scalable Extraction of Training Data from Aligned, Production Language Models. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net. https://openreview.net/forum?id=vjel3nWP2a

[38] D Niezni, H Taub-Tabib, Y Harris, H Sason, Y Amrusi, D Meron-Azagury, M Avrashami, S Launer-Wachs, J Borchardt, M Kusold, A Tiktinsky, T Hope, Y Goldberg, and Y Shamay. 2023. Extending the boundaries of cancer therapeutic complexity with literature text mining. *Artificial Intelligence in Medicine* 145 (Nov. 2023), 102681. doi:10.1016/j.artmed.2023.102681 Epub 2023 Oct 11.

[39] Kyle O'Brien, Stephen Casper, Quentin Anthony, Tomek Korbak, Robert Kirk, Xander Davies, Ishan Mishra, Geoffrey Irving, Yarin Gal, and Stella Biderman. 2025. Deep Ignorance: Filtering Pretraining Data Builds Tamper-Resistant Safeguards into Open-Weight LLMs. arXiv:2508.06601 [cs.LG] https://arxiv.org/abs/2508.06601

[40] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, Sanmi Koyejo, S. Mohamed, A. Agarwal, Danielle Belgrave, K. Cho, and A. Oh (Eds.). http://papers.nips.cc/paper_files/paper/2022/hash/b1efde53be364a73914f58805a001731-Abstract-Conference.html

[41] Annie Palmer. 2025. "FBI investigates Palm Springs bombing, AI chat connection raised". CNBC. https://www.cnbc.com/2025/06/04/fbi-palm-springs-bombing-ai-chat.html.

[42] Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The FineWeb Datasets: Decanting the Web for the Finest Text Data at Scale. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*. https://openreview.net/forum?id=n6SCkn2QaG

[43] Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The FineWeb Datasets: Decanting the Web for the Finest Text Data at Scale. arXiv:2406.17557 [cs.CL] https://arxiv.org/abs/2406.17557

[44] Guilherme Penedo, Hynek Kydlíček, Vinko Sabolčec, Bettina Messmer, Negar Foroutan, Amir Hossein Kargaran, Colin Raffel, Martin Jaggi, Leandro Von Werra, and Thomas Wolf. 2025. FineWeb2: One Pipeline to Scale Them All – Adapting Pre-Training Data Processing to Every Language. arXiv:2506.20920 [cs.CL] https://arxiv.org/abs/2506.20920

[45] Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep Contextualized Word Representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, Marilyn A. Walker, Heng Ji, and Amanda Stent (Eds.). Association for Computational Linguistics, 2227–2237. doi:10.18653/V1/N18-1202

[46] Aleksandra Piktus, Christopher Akiki, Paulo Villegas, Hugo Laurençon, Gérard Dupont, Sasha Luccioni, Yacine Jernite, and Anna Rogers. 2023. The ROOTS Search Tool: Data Transparency for LLMs. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, Danushka Bollegala, Ruihong Huang, and Alan Ritter (Eds.). Association for Computational Linguistics, Toronto, Canada, 304–314. doi:10.18653/v1/2023.acl-demo.29

[47] Audrey Pope. 2024. NYT v. OpenAI: The Times's About-Face. In *Harvard Law Review: Blog Essays*. https://harvardlawreview.org/blog/2024/04/nyt-v-openai-the-timess-about-face/

[48] Alec Radford and Karthik Narasimhan. 2018. Improving Language Understanding by Generative Pre-Training. https://api.semanticscholar.org/CorpusID:49313245

[49] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research* 21, 140 (2020), 1–67. http://jmlr.org/papers/v21/20-074.html

[50] McKenzie Sadeghi and Isis Blachez. 2025. A Well-funded Moscow-based Global 'News' Network has Infected Western Artificial Intelligence Tools Worldwide with Russian Propaganda. NewsGuard. https://www.newsguardtech.com/special-reports/moscow-based-global-news-network-infected-western-artificial-intelligence-russian-propaganda/.

[51] Stefano Schuppli, Fawzi Mohamed, Henrique Mendonça, Nina Mujkanovic, Elia Palme, Dino Conciatore, Lukas Drescher, Miguel Gila, Pim Witlox, Joost VandeVondele, Maxime Martinasso, Thomas C. Schulthess, and Torsten Hoefler. 2025. Evolving HPC services to enable ML workloads on HPE Cray EX. *CoRR* abs/2507.01880 (2025). arXiv:2507.01880 doi:10.48550/ARXIV.2507.01880

[52] Shutterstock. 2025. List of Dirty, Naughty, Obscene, and Otherwise Bad Words. https://github.com/LDNOOBW/List-of-Dirty-Naughty-Obscene-and-Otherwise-Bad-Words. Originally compiled by Shutterstock; GitHub mirror by LDNOOBW, CC BY 4.0 license.

[53] Perplexity AI Team. 2025. Open Sourcing R1 1776, a version of the DeepSeek-R1 model that has been post-trained to provide unbiased, accurate, and factual information. https://www.perplexity.ai/hub/blog/open-sourcing-r1-1776

[54] Simone Ulmer. 2022. At CSCS, energy efficiency is a key priority, even at high performance. https://www.cscs.ch/science/computer-science-hpc/2022/at-cscs-energy-efficiency-is-a-key-priority-even-at-high-performance

[55] Vuk Vuković, Akhil Arora, Huan-Cheng Chang, Andreas Spitz, and Robert West. 2022. Quote Erat Demonstrandum: A Web Interface for Exploring the Quotebank Corpus. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval* (Madrid, Spain) *(SIGIR '22)*. Association for Computing Machinery, New York, NY, USA, 3350–3354. doi:10.1145/3477495.3531696

[56] Wenxuan Wang, Zhaopeng Tu, Chang Chen, Youliang Yuan, Jen-tse Huang, Wenxiang Jiao, and Michael R. Lyu. 2024. All Languages Matter: On the Multilingual Safety of LLMs. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, Lun-Wei Ku, Andre Martins, and Vivek Srikumar (Eds.). Association for Computational Linguistics, 5865–5877. doi:10.18653/V1/2024.FINDINGS-ACL.349

[57] Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned Language Models are Zero-Shot Learners. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net. https://openreview.net/forum?id=gEZrGCozdqR

[58] Arthur Wuhrmann, Andrei Kucharavy, and Anastasiia Kucherenko. 2025. Low-Perplexity LLM-Generated Sequences and Where To Find Them. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, Jin Zhao, Mingyang Wang, and Zhu Liu (Eds.).

Association for Computational Linguistics, Vienna, Austria, 774–783. doi:10.18653/v1/2025.acl-srw.51

[59] Edwin Zhang, Nikhil Gupta, Raphael Tang, Xiao Han, Ronak Pradeep, Kuang Lu, Yue Zhang, Rodrigo Nogueira, Kyunghyun Cho, Hui Fang, and Jimmy Lin. 2020. Covidex: Neural Ranking Models and Keyword Search Infrastructure for the COVID-19 Open Research Dataset. In *Proceedings of the First Workshop on Scholarly Document Processing*, Muthu Kumar Chandrasekaran, Anita de Waard, Guy Feigenblat, Dayne Freitag, Tirthankar Ghosal, Eduard Hovy, Petr Knoth, David Konopnicki, Philipp Mayr, Robert M. Patton, and Michal Shmueli-Scheuer (Eds.). Association for Computational Linguistics, Online, 31–41. doi:10.18653/v1/2020.sdp-1.5

[60] Yiming Zhang, Javier Rando, Ivan Evtimov, Jianfeng Chi, Eric Michael Smith, Nicholas Carlini, Florian Tramèr, and Daphne Ippolito. 2025. Persistent Pre-training Poisoning of LLMs. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*. OpenReview.net. https://openreview.net/forum?id=eiqrnVaeIw