

## JCST Papers

**Only for Academic and Non-Commercial Use**

Thanks for Reading!



[Survey](#)

[Computer Architecture and Systems](#)

[Artificial Intelligence and Pattern Recognition](#)

[Computer Graphics and Multimedia](#)

[Data Management and Data Mining](#)

[Software Systems](#)

[Computer Networks and Distributed Computing](#)

[Theory and Algorithms](#)

[Emerging Areas](#)



JCST WeChat

Subscription Account

JCST URL: <https://jcst.ict.ac.cn>

SPRINGER URL: <https://www.springer.com/journal/11390>

E-mail: [jcst@ict.ac.cn](mailto:jcst@ict.ac.cn)

Online Submission: <https://mc03.manuscriptcentral.com/jcst>

Twitter: JCST\_Journal

LinkedIn: Journal of Computer Science and Technology

# Data Preparation for Large Language Models

Hao Liang<sup>1, 2</sup> (梁昊), *Student Member, CCF*, Zhen Hao Wong<sup>3</sup> (黄钰浩), Rui-Tong Liu<sup>3</sup> (刘芮彤)  
Yu-Han Wang<sup>3</sup> (王宇涵), Mei-Yi Qiang<sup>4</sup> (强美伊), Zheng-Yang Zhao<sup>1</sup> (赵正阳), Cheng-Yu Shen<sup>4</sup> (申程宇)  
Cong-Hui He<sup>5</sup> (何聪辉), Wen-Tao Zhang<sup>1, 2, \*</sup> (张文涛), *Member, CCF, ACM, IEEE*  
and Bin Cui<sup>6, 7, \*</sup> (崔斌), *Fellow, CCF, IEEE*

<sup>1</sup> Center for Data Science, Peking University, Beijing 100871, China

<sup>2</sup> Beijing Zhongguancun Academy, Beijing 100871, China

<sup>3</sup> School of Mathematical Science, Peking University, Beijing 100871, China

<sup>4</sup> School of Software and Microelectronics, Peking University, Beijing 100871, China

<sup>5</sup> Shanghai Artificial Intelligence Laboratory, Shanghai 200232, China

<sup>6</sup> School of Computer Science, Peking University, Beijing 100871, China

<sup>7</sup> Beijing Key Laboratory of Software and Hardware Cooperative Artificial Intelligence Systems, Beijing 100871, China

E-mail: hao.liang@stu.pku.edu.cn; zhenhao1141@stu.pku.edu.cn; BerliewBabson814@gmail.com; fzzh040114@gmail.com  
myqiang25@stu.pku.edu.cn; zhengyangzhao25@stu.pku.edu.cn; 2501210669@stu.pku.edu.cn; heconghui@pjlabor.org.cn  
wentao.zhang@pku.edu.cn; bin.cui@pku.edu.cn

Received September 15, 2025; accepted January 11, 2026.

**Abstract** Large language models (LLMs) have demonstrated remarkable generalization capabilities across diverse domains, largely attributed to the availability of massive amounts of high-quality training data. Recently, the development paradigm of LLMs has been shifting from a model-centric to a data-centric perspective. In this paper, we provide a comprehensive survey of data preparation algorithms and workflows for LLMs, categorized into three stages: pre-training, continual pre-training, and post-training. We further summarize widely used datasets along with their associated data preparation method, offering a practical reference for researchers who may lack extensive experience in the field of data preparation. Finally, we outline potential directions for future work, highlighting open challenges and opportunities in advancing data preparation for LLMs.

**Keywords** data-centric artificial intelligence (AI), data management, large language model (LLM)

## 1 Introduction

Large language models (LLMs) have emerged as a central focus in artificial intelligence (AI), achieving remarkable success across a wide range of applications. While architectural innovations and training strategies have received substantial attention, the role

of data preparation has increasingly proven to be a critical driver for advancing both the performance and generalization ability of LLMs.

As shown in Fig.1, data preparation is the foundation of LLMs. It turns raw text and multimodal data into training-ready signals through four steps: collection, filtering, generation, and evaluation. High-quality

---

Survey

Special Issue: Celebrating the 40th Anniversary of JCST

This work was supported by the National Natural Science Foundation of China under Grant Nos. 92470121, 62402016, U23B2048, and U22B2037, the National Key Research and Development Program of China under Grant No. 2024YFA1014003, Beijing Zhongguancun Academy under Grant Nos. C20250204 and C20250602, the High-Performance Computing Platform of Peking University, and the Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China under Grant No. JYB2025XDXM108.

\*Corresponding Author (Wen-Tao Zhang provided overall supervision and led the conceptualization and research design of the project; Bin Cui oversaw project administration, coordinated the research activities, and contributed substantially to manuscript revision.)

©Institute of Computing Technology, Chinese Academy of Sciences 2026

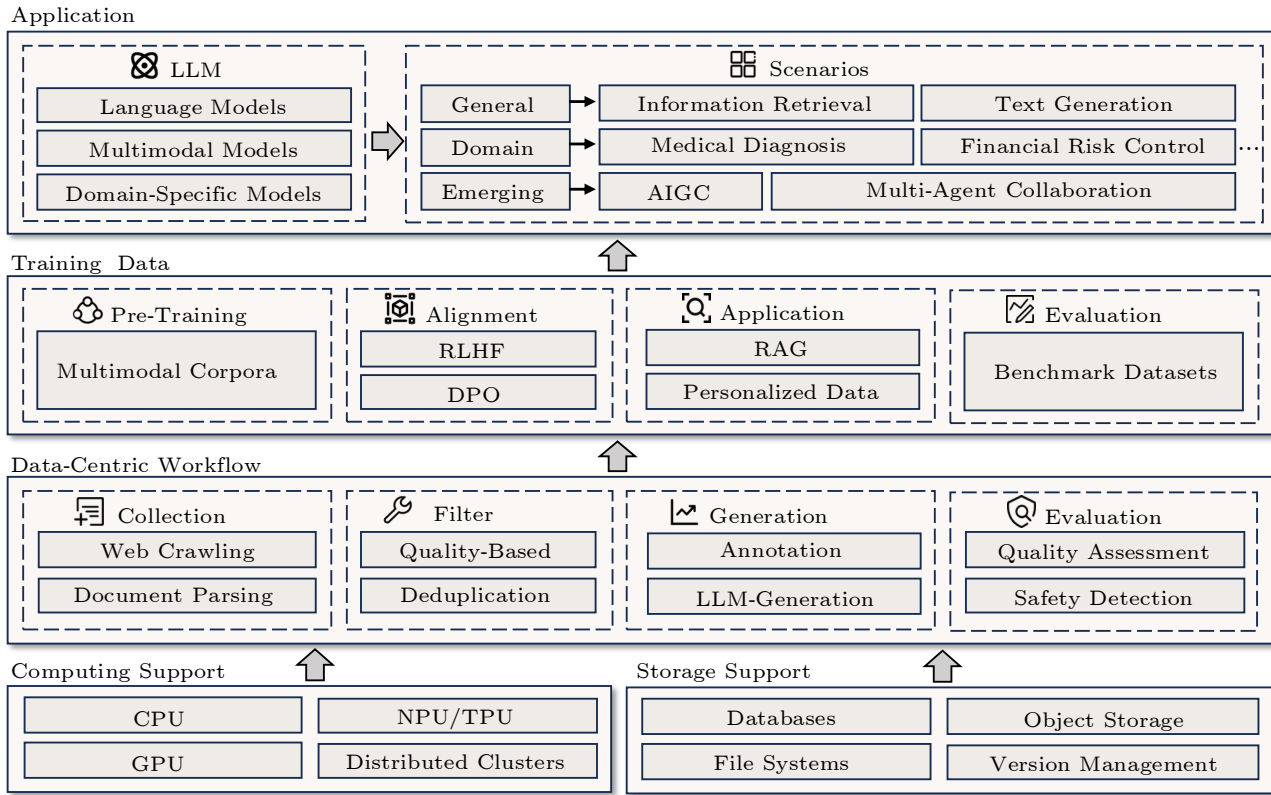


Fig.1. Overview of data for LLMs. AIGC: artificial intelligence generated content, RLHF: reinforcement learning from human feedback, DPO: direct preference optimization, RAG: retrieval-augmented generation, NPU: neural processing unit, TPU: tensor processing unit, GPU: graphics processing unit.

ty, well-targeted data strengthens learning, improves safety, and saves compute; poor data does the opposite. The prepared datasets then drive pre-training, alignment (reinforcement learning from human feedback (RLHF), direct preference optimization (DPO)), and application tuning (retrieval-augmented generation (RAG), personalization), with benchmarks closing the loop. In short, stronger data pipelines lead to stronger, more reliable LLMs across domains. Several recent surveys have attempted to summarize data preparation approaches<sup>[1-6]</sup>. These researches have provided valuable insights and comprehensive overviews of existing methods, establishing a strong foundation for subsequent research. Nevertheless, they face several limitations that highlight the need for an updated and more focused treatment of the topic.

1) *Insufficient Focus on Data Preparation*. Existing surveys are not exclusively devoted to the systematic study of data preparation<sup>[1-6]</sup>; instead, they discuss it only as a secondary component within broader topics.

2) *Limited Timeliness and Coverage of Recent Advances*. Data preparation operators and workflows are evolving rapidly. For instance, Bai *et al.*<sup>[1]</sup> primar-

ily concentrated on multimodal aspects, Zha *et al.*<sup>[2]</sup> offered a broad overview of data-centric AI but without a dedicated focus on data preparation and their work is now relatively dated, while Zhou *et al.*<sup>[3]</sup> did not adequately capture the latest trends in data workflows.

3) *Absence of Systematic Data Workflow Summarization*. Prior surveys<sup>[1-6]</sup> rarely offer a structured abstraction of data preparation workflows. As a result, the relationships between individual operators, pipeline stages, and training paradigms remain fragmented and difficult to compare or reuse in practice.

In this work, we aim to bridge these gaps by presenting a comprehensive and up-to-date survey of text data preparation for LLMs. We systematically organize existing methods from three complementary training perspectives: pre-training, continual pre-training, and post-training. For each stage, we summarize recent advances in awesome datasets and their corresponding data preparation techniques, operator patterns, and abstract reusable data workflows, providing a unified view that connects data preparation practices with model training objectives.

Our contributions are threefold.

1) We present a comprehensive and up-to-date summary of the latest data preparation algorithms designed for LLMs.

2) We propose a new taxonomy that begins from novel data operators and extends to commonly used workflows, systematically illustrating how operators can be composed and tailored for LLM training.

3) We compile and categorize widely used datasets along with their preparation workflows, providing a practical reference to help both newcomers and practitioners rapidly navigate and implement data preparation strategies.

As shown in Fig.2, this paper is organized as follows. Section 2 reviews data preparation for pre-training (PT), and Section 3 extends this to continual pre-training (CPT). Section 4 details post-training data preparation. Section 5 surveys commonly used datasets together with their end-to-end preparation workflows. Finally, Section 6 outlines open challenges and future research directions for data preparation in LLMs.

## 2 Pre-Training Data Preparation

### 2.1 Data Collection

Pre-training data collection for LLMs has evolved through three main phases: scale-oriented crawling, reproducibility-oriented mixtures, and capability-oriented enrichment. This reflects a shift from collecting everything to curating responsibly, with increasing at-

tention to reproducibility, legal compliance, and domain specialization.

*Scale-Oriented Crawling.* Early scale-oriented crawling pipelines prioritize the size and diversity, often harvesting trillions of tokens from Common Crawl snapshots<sup>①</sup>. A representative example is CCNet<sup>[7]</sup>, which introduces language identification, boilerplate removal, and perplexity-based cleaning for multilingual corpora. Subsequent corpora extend this paradigm: WebText<sup>[8]</sup> leverages Reddit upvotes as a proxy for link quality; C4<sup>[9]</sup> applies large-scale filtering and deduplication; ClueWeb22<sup>[10]</sup> adds browser rendering, visual DOM features, and semantic annotations to better extract page content. More recent multilingual resources further improve coverage for low-resource languages, such as CulturaX<sup>[11]</sup> and GlotCC<sup>[12]</sup>. Despite the corpora’s unprecedented scale, this stage suffers from noisy quality, uneven domain coverage, and limited reproducibility.

*Reproducibility-Oriented Mixtures.* RedPajama-Data V2<sup>[13]</sup> aggregates more than 80 Common Crawl snapshots and provides a deduplicated pool with more than 40 document-level quality signals. Dolma<sup>[14]</sup> provides a 3-trillion (3T)-token open corpus plus a modular curation toolkit. FineWeb and FineWeb-Edu<sup>[15]</sup> report strong gains on knowledge- and reasoning-intensive benchmarks with lightweight, scalable filters. Common Corpus<sup>[16]</sup> prioritizes permissively licensed/-public-domain sources at trillion-token scale to mitigate legal risk. DataComp-LM<sup>[17]</sup> standardizes a 240T-token corpus, pre-training recipes, and a broad evaluation suite to enable controlled, data-centric compar-

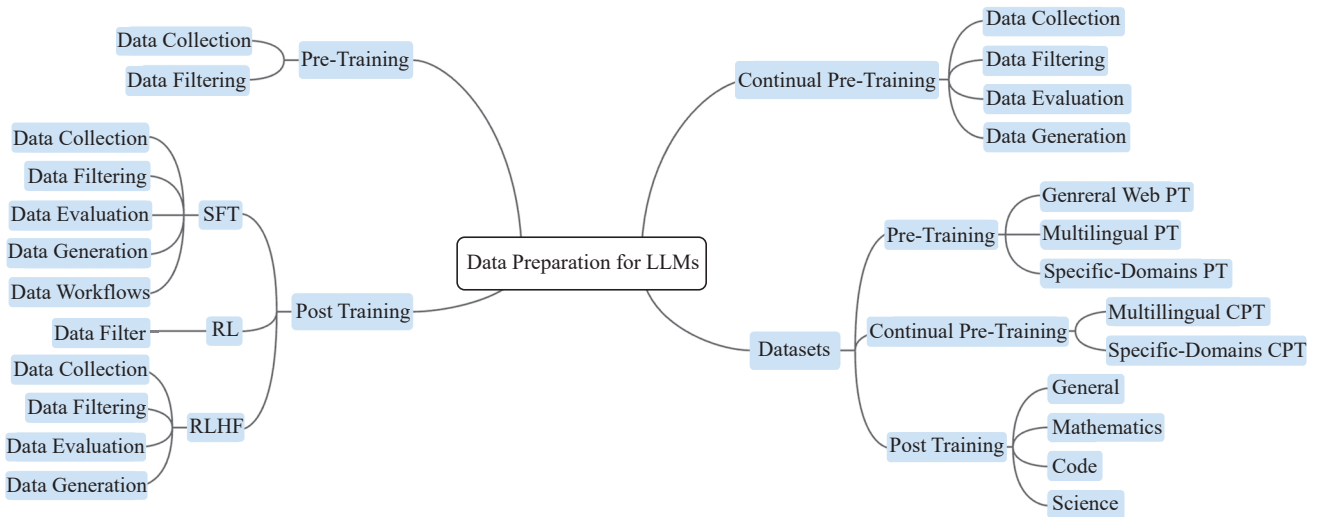


Fig.2. Overview of data preparation for LLMs. SFT: supervised fine-tuning, RL: reinforcement learning, PT: pre-training, CPT: continual pre-training.

<sup>①</sup><https://commoncrawl.org/>, Jan. 2026.

isons. Collectively, this stage shifted the community from data at scale to open and composable mixtures, with reproducibility and transparency as guiding principles.

*Capability-Oriented Enrichment.* This extends corpora for targeted domains and specialized skills. For programming, the Stack v2 and StarCoder2<sup>②</sup> expand coverage to more than 600 languages and show strong code/reasoning results. For mathematics, OpenWebMath<sup>①</sup> preserves LaTeX at scale, OpenMathInstruct-2<sup>③</sup> contributes large problem-solution pairs, and MegaMath<sup>④</sup> scales to 371 billion (B) mathematics (math) tokens via web/code/synthetic integration with extensive ablations. In biomedical domains, corpora emphasize curated literature and clinical text: PMC-LLaMA<sup>⑤</sup> continually pre-trains on PubMed Central; Me-LLaMA<sup>⑥</sup> releases a 129B-token biomedical corpus plus instruction data; Biomed-Enriched<sup>⑦</sup> applies LLM-based selection over PubMed. Overall, targeted enrichment improves efficiency and specialization while introducing risks around overfitting, licensing, and benchmark contamination.

*Data Parsing for More Information.* OCR books provide coherent long-form discourse and rich document structure (chapters, figures, tables, equations), making them valuable for pre-training. Tools such as MinerU enable high-precision PDF-to-Markdown and JSON conversion with integrated layout parsing and OCR, retaining math and code blocks and reading order to boost long-context and domain skills while controlling contamination risk<sup>⑧</sup>.

## 2.2 Data Filtering

A fundamental step in data preparation is to filter inappropriate, redundant, or low-quality content before it enters the pre-training corpus. Large-scale crawled data may contain sensitive information, offensive expressions, social biases, duplicated documents, evaluation set leakage, or undesirable content such as emojis and non-textual artifacts. Filtering methods have thus been extensively developed along four primary dimensions: 1) sensitive and harmful content, 2) bias and fairness, 3) redundancy and contamination, and 4) language, format, and quality control.

*Sensitive and Harmful Content.* This category encompasses personally identifiable information (PII) and toxic or offensive languages. PII refers to any da-

ta linked to a specific individual, such as names, addresses, emails, or phone numbers<sup>⑨</sup>. Heuristic approaches based on regular expressions and dictionary matching are widely adopted<sup>⑩</sup>, while named entity recognition (NER) models such as Presidio<sup>⑪</sup>, or BERT-based NER systems trained on annotated resources like WikiPII<sup>⑫</sup>, further improve detection accuracy. Toxic content, often defined as rude or disrespectful expressions that may discourage participation<sup>⑬</sup>, is filtered using sensitive word lists<sup>⑭</sup>,  $n$ -gram classifiers<sup>⑮</sup>, webpage-level URL blocklists<sup>⑯</sup>, or model-based classifiers such as FastText<sup>⑰</sup> and BERT-based toxicity detectors<sup>⑱</sup>.

*Bias and Fairness.* Filtering targets social biases present in training corpora<sup>⑲</sup>. Counterfactual data augmentation (CDA)<sup>⑳</sup> mitigates bias by swapping attribute words (e.g., “he”/“she”), while variants such as counterfactual data substitution (CDS)<sup>㉑</sup>, blindness replacement<sup>㉒</sup>, and selective augmentation<sup>㉓</sup> further refine CDA’s effect. Beyond augmentation, scoring documents with pre-trained language models conditioned on trigger phrases<sup>㉔</sup> enables the exclusion of biased samples.

*Redundancy and Contamination.* Filtering ensures pre-training efficiency, data diversity, privacy or security, and fairness in evaluation. Contamination filtering prevents training-test overlap, which could bias evaluation results. Exact string and substring matching are widely used<sup>㉕</sup>, while  $n$ -gram overlap criteria are adopted by large models: GPT-3 applies 13-gram matching<sup>㉖</sup>, PaLM enforces 70% overlap in 8-grams<sup>㉗</sup>, and LLaMA2 flags contamination when any  $>10$ -token  $n$ -gram overlaps between training and evaluation data<sup>㉘</sup>. These strategies ensure that evaluation is not compromised by memorized training content. Deduplication identifies and removes redundant data to improve pre-training quality, efficiency, generalization, and privacy/security<sup>㉙</sup>. Three paradigms prevail. Exact-based methods detect identical spans via whole-document hashing (e.g., MD5/SHA; used in CCNet<sup>㉚</sup>, LLaMA<sup>㉛</sup>, The Stack<sup>㉜</sup>; with Bloom filters for scale in Dolma<sup>㉝</sup> and RedPajama-V2<sup>㉞</sup>) and exact substring matching (the C4 rule of three consecutive sentences<sup>㉟</sup>; EXACTSUBSTR<sup>㊱</sup> with suffix arrays<sup>㊲</sup> removing more than 50 token duplicates, adopted by RefinedWeb<sup>㊳</sup> and clinical corpora<sup>㊴</sup>). Fuzzy-based methods remove near-duplicates using similarity-MinHash<sup>㊵</sup> (GPT-3<sup>㊶</sup>, The Pile<sup>㊷</sup>, Go-

②<https://huggingface.co/datasets/bigcode/the-stack-v2>, Jan. 2026.

③<https://www.presidio.com/>, Jan. 2026.

pher<sup>[61]</sup>, RefinedWeb<sup>[33]</sup>, SlimPajama<sup>[62]</sup>, RedPajama-V2<sup>[55]</sup>, SimHash<sup>[63]</sup> (RedPajama-1T<sup>[4]</sup>), plus OSCAR-23.01<sup>[5]</sup> (TLSH), TigerBot<sup>[64]</sup> (TF-IDF), and PaLM<sup>[49]</sup> (Levenshtein for code). Recent advances include SoftDedup<sup>[65]</sup> (redundancy reweighting), LSHBloom<sup>[66]</sup> (faster web-scale MinHashLSH), limits exposed by the mosaic memory study<sup>[67]</sup>, and practical pipelines like FineWeb<sup>[15]</sup>. Model-based methods embed texts and cluster to flag semantic duplicates-SemDeDup<sup>[68]</sup> ( $k$ -means + cosine), extended by D4<sup>[69]</sup> (second-stage pruning near centroids) with implementations in MiniPile<sup>[70]</sup> and DataJuicer<sup>[71]</sup>; others fine-tune encoders include the Noise-Robust method<sup>[72]</sup> with SBERT<sup>[73]</sup> retrieval + cross-encoder re-ranking, and generative approaches like GenDedup<sup>[74]</sup> that flag duplicates via confident keyword prediction as a memorization signal.

*Language, Format, and Quality Filters.* These filters ensure that the pre-training corpus consists of clean, usable, and consistent data. Large-scale crawled datasets often contain non-textual elements such as emojis, HTML boilerplate, embedded scripts, undesired code snippets, malformed characters, truncated documents, and content in unsupported languages. To address these issues, preprocessing pipelines combine heuristic rules, language classifiers, and format heuristics. Language identification tools, including fastText LID<sup>[6]</sup>, langid.py<sup>[7]</sup> implementations, are widely used to retain only documents in the target language, often applying confidence thresholds to balance coverage and precision<sup>[7, 26]</sup>. Format cleaning typically involves stripping HTML tags, scripts, and other boilerplate content, normalizing whitespace and punctuation, and removing lines with excessive symbols or non-natural text. This is accomplished using robust extraction tools such as Trafilatura<sup>[75]</sup> and Boilerpipe<sup>[76]</sup> as well as dataset-specific pipelines<sup>[7, 33]</sup>. Quality control further filters documents based on simple structural features, such as length, alphabetic-content ratio, and symbol density, and on model-based signals, including perplexity thresholds or classifier scores. These measures eliminate extremely short, malformed, or otherwise noisy texts that could hinder model training. Such comprehensive preprocessing practices are standard across modern large-scale pre-training pipelines, including T5<sup>[9]</sup>, CCNet<sup>[7]</sup>, RefinedWeb<sup>[33]</sup>, The Pile<sup>[60]</sup>, RedPajama<sup>[13]</sup>, SlimPajama<sup>[62]</sup>, FineWeb<sup>[15]</sup> series, and Dolma<sup>[14]</sup>.

ma<sup>[62]</sup>, FineWeb<sup>[15]</sup> series, and Dolma<sup>[14]</sup>.

### 3 Continual Pre-Training Data Preparation

Continual pre-training (CPT) is typically interposed between base pre-training and fine-tuning to incorporate new knowledge (domains, languages, and up-to-date facts) without retraining from scratch. In practice, this means further training a pre-trained LLM on new, unlabeled corpora drawn from specific sources such as recent publications, domain-specific documents, or low-resource languages. This process enriches the model with fresh content while preserving general language understanding. In the subsequent subsections of this section, we review CPT-specific data preparation strategies under the categories of collection, filtering, evaluation, and generation.

#### 3.1 Data Collection

*Domain-Specific Corpora.* Data collection emphasizes freshness and complementarity, targeting areas where the base model remains undertrained. One major direction is the collecting of domain-specific corpora, where large unlabeled collections such as medical research papers, legal documents, or financial news are gathered to bridge domain gaps. In finance, FinPythia-6.9B<sup>[77]</sup> leverages targeted selection from financial news, reports, and filings, while the Japanese financial CPT corpus<sup>[78]</sup> is built from official policy statements and institutional publications.

*Language Coverage.* Another major line of work emphasizes language coverage, aiming to extend LLMs to under-represented or low-resource languages. Zilio *et al.* built a 120B-token Portuguese corpus from web crawls and showed that continued pre-training from an English-centric base model significantly boosts performance in Portuguese<sup>[79]</sup>. Similarly, multilingual CPT projects such as Glot500<sup>[80]</sup>, EMMA-500<sup>[81]</sup>, Sailor<sup>[82]</sup>, and AURORA-M<sup>[83]</sup> extend coverage to low-resource and under-represented languages through large-scale curation, filtering, and balancing of multilingual corpora, while complementary efforts in efficient CPT for low-resource languages<sup>[84]</sup> emphasize vocabulary adaptation and selective corpus expansion to achieve strong gains with limited data.

<sup>[4]</sup><https://github.com/togethercomputer/RedPajama-Data/>, Jan. 2026.

<sup>[5]</sup><https://huggingface.co/datasets/oscar-corpus/OSCAR-2301>, Jan. 2026.

<sup>[6]</sup><https://fasttext.cc/docs/en/language-identification.html>, Jan. 2026.

<sup>[7]</sup><https://pypi.org/project/langdetect/>, Jan. 2026.

*Temporal Freshness.* A third emphasis is on temporal freshness, which ensures that models acquire new knowledge aligned with evolving facts and contexts. For example, TemporalWiki<sup>[85]</sup> is applied on the incremental differences between consecutive Wikipedia dumps, showing that training on the incremental corpus achieves comparable perplexity improvements with far less data. Similarly, Nemotron-CC<sup>[86]</sup> leverages the most recent Common Crawl snapshots and applies aggressive filtering to refine long-horizon pre-training datasets, while NeuScraper<sup>[87]</sup> introduces neural web scraping techniques to extract clean and up-to-date web content. Collectively, these efforts highlight that data collection for CPT prioritizes newness-capturing fresh domains, novel content, and the latest information to keep LLMs up to date and contextually relevant.

### 3.2 Data Filtering

*Long-Context Filters.* In CPT, data filtering serves distinct objectives compared with the initial pre-training stage, as it must adapt corpora to evolving model capabilities and target applications. One emerging direction is long-context filters, which focuses on identifying documents with coherent long-range dependencies that can support training for extended context windows.

*Language and Genre Filters.* Another important line of work applies language and genre filters to ensure alignment with the intended domain or task. For instance, in the construction of a large Portuguese corpus, classifiers were trained to detect domain-specific categories such as science, technique, engineering, math (STEM), education, and toxic content, thereby retaining only relevant text segments<sup>[79]</sup>. Similarly, code-focused CPT isolates pure programming language repositories by filtering out natural language, ensuring that the model adapts specifically to software-related tasks. Such strategies highlight the importance of carefully segmenting data according to linguistic or stylistic attributes so that models are exposed to the most relevant distributions.

*Quality Filters.* Finally, quality filtering in CPT largely follows the same technical route as in the pre-training stage, relying on standard methods such as rule-based cleaning, de-duplication, and lightweight classifiers (e.g., FastText) to remove noisy, redundant, or irrelevant text<sup>[84, 88]</sup>. However, a distinguishing feature of CPT lies in the use of model-informed filtering, where checkpoints from near-converged

LLMs or smaller proxy models are employed to validate the utility of new data. For instance, the Ultra-FineWeb pipeline uses a near-final LLM checkpoint to score candidate batches, while simple classifiers iteratively prune low-value segments, resulting in higher corpus quality and downstream gains<sup>[89]</sup>.

### 3.3 Data Evaluation

*Similarity-Based Scoring.* Evaluating the utility of new data is crucial to ensure that added corpora meaningfully enhance model performance rather than introducing redundant or irrelevant information. One common approach is similarity-based scoring, which quantifies how new data relates to existing pre-training corpora or downstream tasks<sup>[90–93]</sup>. Distributional measures, such as token or embedding similarities, can provide a coarse estimate of corpus alignment. Although similarity measurements offer only rough guidance<sup>[94]</sup>, large-scale experiments reveal consistent patterns: measuring inter-corpus similarity through embeddings or topic modeling can help design the curriculum for CPT. Yildiz *et al.*<sup>[95]</sup> further demonstrated that the semantic ordering of domain corpora significantly affects continual learning outcomes, successively related domains strengthen specialization, while interleaving diverse domains promotes broader transfer. These findings confirm that strategic similarity assessment is valuable for developing effective CPT curricula.

*Proxy Model Evaluation.* A complementary strategy is proxy model evaluation, where candidate data is directly tested through light training or small-scale pilot experiments. Wang *et al.*<sup>[96]</sup> proposed an efficient verification procedure in which a near-converged checkpoint is exposed to new batches of data during the final training epochs, and improvements on held-out loss are monitored to quickly assess corpus utility. This line of work highlights the importance of model-informed evaluation, as opposed to relying solely on generic quality metrics. Recent studies have demonstrated the broader value of such proxy-based evaluation. Recent studies highlight the broader value of proxy-based evaluation in continual pre-training. Chen *et al.*<sup>[97]</sup> further demonstrated that lightweight validation loops reliably predict long-term gains, while Xie *et al.*<sup>[77]</sup> confirmed their utility in uncovering domain-specific corpora with outsized impact. Domain-oriented efforts such as AstroMLab 2<sup>[98]</sup> underscore the necessity of proxy validation when curating highly specialized scientific data. Collectively, these re-

searches establish proxy model validation as a principled and efficient mechanism to align new corpora with both the evolving state of LLMs and task-specific objectives.

### 3.4 Data Generation

*Raw Text Reformatting.* Synthetic data generation has emerged as a key strategy to enhance continual pre-training corpora, providing additional reasoning signals, instructions, and domain knowledge beyond what is available in raw text collections. One important category is raw text reformatting, which restructures existing corpora into supervision-friendly formats such as question-answer pairs or instruction-style prompts. For instance, paraphrasing and stylistic rewrites of large-scale web data have been proven to substantially improve data efficiency<sup>[97, 99, 100]</sup>. Similarly, YuLan-Mini<sup>[101]</sup> and the Phi-4<sup>[102]</sup> highlight the value of dialogue reformulation and mixed-style conversions in creating compact yet versatile corpora that retain knowledge coverage while reducing redundancy. Collectively, these studies indicate that reformatting strategies, such as style conversion, language adaptation, and dialogue recasting, play a crucial role in making raw text more suitable for continual pre-training and better aligned with downstream application requirements.

*Reasoning Ability Augmentation.* A second line of work explicitly targets reasoning ability augmentation by injecting synthetic thought processes into training corpora. Reasoning CPT<sup>[103]</sup> prompts LLMs to reconstruct hidden reasoning steps behind STEM and law texts, leading to consistent improvements on multi-domain reasoning benchmarks. Similarly, continual pre-training with entity-linked corpus expansion in EntiGraph<sup>[104]</sup> provides richer semantic structures that foster reasoning over inter-entity relations. MIND<sup>[105]</sup> generates synthetic dialogues with knowledge gaps to strengthen problem-solving ability. These methods demonstrate that injecting explicit reasoning behavior through synthetic data is an effective strategy for deepening inference capacity in continual pre-training.

*Domain Expansion.* Furthermore, domain expansion methods have been increasingly used in CPT to overcome sparse domains or task corpora by synthesizing or selecting additional relevant data. For instance, FinPythia-6.9B<sup>[77]</sup> extends financial corpora through targeted continual pre-training, while MachineLearningLM<sup>[106]</sup>, MIND<sup>[105]</sup>, and the AstroMLab

series<sup>[98]</sup> enrich scientific and mathematical domains; across these disciplines, domain-focused CPT leverages carefully filtered or augmented corpora to achieve substantial gains in specialized reasoning and knowledge-intensive tasks. TRAIT<sup>[107]</sup> applies a task-oriented in-domain augmentation framework, selecting domain-matched examples from general corpora and then generating synthetic passages that guide the model on how to apply domain knowledge to downstream tasks like advertisement.

## 4 Post-Training Data Preparation

After the pre-training stage, LLMs possess broad world knowledge and general linguistic competence, yet they often lack alignment with specific tasks and human-preferred behaviors. This gap underscores the critical importance of post-training data preparation, particularly for supervised fine-tuning (SFT). Unlike the pre-training phase, which emphasizes scale and diversity, SFT relies on carefully curated, task-relevant, and instruction-oriented datasets that refine model behavior toward reliability, safety, and usability. Recent studies have highlighted that high-quality post-train data can significantly improve instruction-following ability, mitigate biases inherited from pre-training corpora, and enhance robustness across diverse downstream scenarios. Therefore, the preparation of post-train data should not merely focus on quantity, but rather prioritize coverage of task diversity, balance of domain distributions, and stringent quality control. In this sense, post-train data serves as the foundation for transforming a knowledge-rich but unaligned model into a task-effective, human-aligned assistant.

### 4.1 SFT

Supervised fine-tuning (SFT) has become a key step in aligning pre-trained models during the post-training phase. The primary objective of SFT is to perform supervised learning on high-quality (instruction, output) pairs, thereby can teach the model how to understand and follow human instructions.

To facilitate effective supervised fine-tuning, diverse datasets that align with the target domain and task are used. These datasets typically include a combination of instruction-response pairs and text-only SFT data. By exposing the LLMs to this rich variety of data, the model can develop a comprehensive understanding of the target domain and acquire the nec-

essary skills to perform the desired tasks effectively.

The preparation of supervised fine-tuning datasets usually includes the following steps.

- Data collection: acquire raw data from diverse sources.
- Data filtering: remove low-quality, redundant, or irrelevant samples while retaining high-quality data.
- Data evaluation: assess data diversity, authenticity, and task relevance using manual or automated metrics.
- Data generation: expand and reformulate data according to different task types, converting them into instruction-response pairs.
- Data preparation workflows: integrate and automate the above steps to build standardized data preparation pipelines.

#### 4.1.1 Data Collection

We mainly collect data from the following main categories: existing corpora, manually-constructed data, and LLM generated data.

*Existing Corpus.* A “ready-to-use” approach of constructing data is to collect open-source traditional natural language process (NLP) datasets<sup>[108–110]</sup>. This method entails minimal overhead and ensures an ample quantity of data. Aside from gathering open-source traditional NLP datasets, there is also a plethora of real-world dialog examples for LLMs on high-quality web-based QA communities. LIMA<sup>[111]</sup> samples 75% of its instructions from two high-quality web-based QA communities (Stack Exchange, wikiHow).

*Manually-Constructed Data.* Free Dolly<sup>®</sup> contains 15 000 pieces of manually constructed English instruction data, which is directed to refrain from utilizing external AI systems or sourcing information from web sources. Ouyang *et al.*<sup>[112]</sup> curated an SFT dataset (13 000 training prompts) with labeler demonstrations. Open-Assistant<sup>[113]</sup> contains 35 different languages, and consists of 161 443 conversation data distributed across 66 497 conversation trees. Aya Dataset<sup>[114]</sup> collects and annotates via the Aya Annotation Platform. OL-CC<sup>®</sup> is the first open-source Chinese instruction dataset generated through crowdsourcing and manual efforts.

*LLM Generated Data.* As LLMs increasingly demonstrate conversational and text generation capabilities on par with human performance, a growing line of work leverages their outputs as raw data for SFT. This approach typically collects user-LLM interaction logs. For example, Vicuna<sup>®</sup> gathers around 70 000 conversations from ShareGPT.com, where users can share their ChatGPT interactions. Instruction Wild<sup>®</sup> collects (110K in the v2 dataset, 429 in the v1 dataset) instructions from ChatGPT usage sharing and released both English and Chinese versions. Recent research has moved toward more refined strategies that hybridize LLMs and real-world data to balance scale and authenticity. WebR<sup>[115]</sup> employs a web reconstruction approach to automatically identify potential (instruction, response) pairs from raw webpages, which are then rewritten and expanded by strong generative models. BARE<sup>[116]</sup> uses a dual-model strategy where a base model ensures diversity and an instruction-tuned model enhances authenticity.

#### 4.1.2 Data Filtering

After the data collection, the data filtering section aims to ensure that the corpus is clean, diverse, and suitable for supervised fine-tuning. Some filtering operations are shared with the pre-training stage. Beyond these general approaches, SFT requires task-specific filtering methods that directly target the quality of (instruction, response) pairs. Repairity<sup>[117]</sup> systematically filters out low-quality reasoning traces and retains only high-quality samples for SFT. He *et al.*<sup>[118]</sup> found that the step-wise error rate in complex reasoning tasks significantly affects model performance, suggesting that data filtering should remove samples with high logical error rates or unreliable intermediate steps. TFP<sup>[119]</sup> filters out SFT samples with large contextual discrepancies during the packing stage, significantly improving the efficiency and effectiveness of the fine-tuning process.

#### 4.1.3 Data Evaluation

SFT is essential for adapting LLMs with downstream applications. Studies<sup>[120–122]</sup> showed that even

<sup>®</sup><https://aigloballab.com/free-dolly-introducing-the-worlds-first-truly-open-instruction-tuned-llm/>, Jan. 2026.

<sup>®</sup><https://opendatalab.com/OpenDataLab/OL-CC>, Jan. 2026.

<sup>®</sup><https://lmsys.org/blog/2023-03-30-vicuna/>, Jan. 2026.

<sup>®</sup><https://github.com/XueFuzhao/InstructionWild>, Jan. 2026.

a small set of high-quality instructions can yield strong instruction-following performance. To further enhance efficiency, data selection methods have been explored, mainly falling into four categories: coreset-based, LLM-based, gradient-based, and self-instruction based approaches.

*Coreset-Based Methods.* Coreset-based approaches evaluate data quality by selecting a small yet representative subset that reflects the overall distribution, thereby identifying which samples are most informative for SFT. Early work applied geometric and clustering perspectives, such as the greedy  $k$ -center algorithm<sup>[123]</sup>, to measure representativeness in the feature space. Subsequent methods introduce submodular optimization to maintain diversity and reduce redundancy<sup>[124, 125]</sup>, or adopt sensitivity-based importance sampling to capture the contribution of each data point to model performance<sup>[126, 127]</sup>. More recent work combines task relevance with influence-based evaluation, proposing data-driven selection strategies tailored for instruction tuning<sup>[121, 122]</sup>.

*LLMs-Based Methods.* LLMs have been increasingly used as evaluators to quantify the quality, difficulty, and reasoning reliability of SFT data. <sup>[128]</sup> demonstrates that generative LLMs can serve as robust scoring functions for generated text, enabling customizable evaluation with prompting and in-context examples. Subsequent studies incorporate LLM-derived judgements as evaluation signals for instruction data, either directly as scores or as annotations of complexity and validity, such as MoDS<sup>[129]</sup> and SelectLLM<sup>[130]</sup>. <sup>[131]</sup> explores how LLM-based rewriting and self-refinement can offer an indirect evaluation of the quality of instruction data. Recent work<sup>[132]</sup> has highlighted potential biases when using LLMs as evaluators, as LLMs may prefer responses that resemble their own stylistic patterns, which can undermine objective evaluation. To address this, studies have proposed more robust prompting designs to improve reliability. <sup>[118]</sup> introduces an evaluation method for step-wise reasoning traces, showing that intermediate error rates measured by LLMs are predictive of model performance. <sup>[133]</sup> provides a comprehensive survey of reasoning-trace evaluation, highlighting how LLM-based assessments can be integrated with complementary signals.

*Gradient-Based Methods.* Gradient-based evaluation methods aim to measure the contribution of samples by analyzing their effect on model gradients. The EL2U score<sup>[134]</sup> quantifies how removing a single data

point alters the gradient, though it is limited to single-task scenarios. <sup>[135]</sup> extends this approach to multi-task NLP by computing EL2U for each task and aggregating the scores with an L2 norm. <sup>[136]</sup> adapts influence function formulations to work with the Adam optimizer, storing compressed gradient features in a gradient datastore for efficient evaluation. Recent research has further improved scalability: LoGra<sup>[137]</sup> uses low-rank projection to reduce computational and memory overhead. QLESS<sup>[138]</sup> combines gradient quantization and random projection within the LESS framework to compress gradient representations while preserving evaluation accuracy.

*Self-Instruction Based Methods.* For self-instruction-based methods, the evaluation and selection of data do not require the involvement of any external models. <sup>[139]</sup> proposes a self-instruction method for LLMs to autonomously identify and select challenging data. <sup>[140]</sup> identifies high-quality data as those that, when included as part of a few-shot example, can enhance the model’s performance. <sup>[141]</sup> proposes a novel task-level uncertainty metric that measures the sensitivity of LLMs to instruction perturbations for a task, to identify high-quality data, and combines this with the concept of active learning to iteratively select data. SelectIT<sup>[142]</sup> scores data at three granular levels: token, sentence, and model. Each level’s scoring is calculated based on the previous level, with the model-level score serving as the final score.

#### 4.1.4 Data Generation

This stage aims to generate high-quality (instruction, response) pairs. Depending on the requirements of different downstream tasks, data generation can be categorized into the following five types of tasks.

*Generation and Creative Tasks.* These datasets aim to enhance the model’s ability in text generation and creative writing. The key idea is to transform existing structured text pairs into instruction-based formats. For example, summarization tasks such as CNN/DailyMail<sup>[143]</sup> and XSum<sup>[144]</sup> convert (article, summary) pairs into (article, prompt, summary) triplets, with the summary serving as the target output. To increase creativity, community-sourced data, such as Reddit stories are often reformulated into writing or continuation prompts<sup>[145, 146]</sup>.

*Question Answering and Information Extraction.* The goal of this type is to train models as reliable tools for knowledge retrieval and information extrac-

tion. Extractive QA datasets such as SQuAD<sup>[147]</sup> restructure (context, question, answer) triplets into (given passage, question, answer) triplets. For open-domain QA, datasets like Natural Questions<sup>[148]</sup> and TriviaQA<sup>[149]</sup> directly use (question, answer) pairs as instructions and outputs. Information extraction tasks (e.g., relation extraction) are reformulated into generative instructions, aligning them with the unified framework<sup>[150, 151]</sup>.

*Reasoning and Problem-Solving.* These datasets aim to improve complex reasoning skills, with a focus on providing both the answer and the reasoning process. Chain-of-Thought (CoT)<sup>[152]</sup> explicitly annotates step-by-step reasoning rather than only the final answer. Math datasets such as GSM8K<sup>[153]</sup> and MATH<sup>[154]</sup> construct CoT annotations via manual or semi-automatic methods. Code generation tasks reformulate the problem into: write a program to solve this problem- executable code, with datasets including HumanEval<sup>[155]</sup> and MBPP<sup>[156]</sup>. Further approaches such as PAL<sup>[157]</sup> and PoT<sup>[158]</sup> extend this idea by transforming reasoning chains into programmatic processes, where the instruction is the problem and the response is code with explanations. Logical and common reasoning tasks use datasets like StrategyQA<sup>[159]</sup>. The latest research no longer merely satisfies itself with the correctness of the final answer, but pays more attention to the quality of the reasoning process. BiRM<sup>[160]</sup> proposes a process supervision method with bidirectional reward signals, which also predicts the probability of future success. Diverse Chains of Thought (DCoT)<sup>[161]</sup> proposed generating multiple distinct chains of thought within a single reasoning process, allowing the model to continuously self-correct. This thereby enhances the robustness of the model in the tasks.

*Natural Language Understanding and Classification.* The central approach here is to convert traditional datasets into the unified (instruction, response) format. Pioneering work such as Flan<sup>[109]</sup> and T0<sup>[162]</sup> systematically reformulated dozens of classic tasks (many from GLUE<sup>[163]</sup> and SuperGLUE<sup>[164]</sup>) into instruction-based tasks. Extensive experiments<sup>[165, 166]</sup> have shown that fine-tuning on diverse instructions significantly improves zero-shot generalization on unseen tasks.

*Dialogue and Conversational AI.* Dialogue datasets are constructed by structuring real conversational data into (instruction, response) pairs. For example, ShareGPT collects user-ChatGPT interactions and re-

formats them as (user input, assistant reply), which has been widely adopted by open-source models such as Vicuna<sup>[167]</sup>. More advanced alignment strategies, such as InstructGPT<sup>[112]</sup>, first performing SFT on instruction data, then collecting human-ranked multiple responses. The initial SFT data combined with human-annotated preference data forms the foundation of high-quality conversational ability.

#### 4.1.5 SFT Data Preparation Workflows

In SFT, different workflows may pursue different objectives, but they generally rely on a shared set of fundamental operators, which form the basic building blocks of all pipelines. Common operators include: template-based conversion (mapping existing text pairs into a unified format), data cleaning (language identification, normalization, noise removal), deduplication (exact, fuzzy, or embedding-based), quality filtering (rule-based, lightweight classifiers, or LLM-as-a-judge), safety and privacy filtering (toxic content detection), and metadata and provenance logging (recording source, version, and processing history)<sup>[48, 109, 162, 168–173]</sup>. In practice, SFT preparation can be broadly categorized into the following representative workflows.

- *Corpus-to-Instruction Workflow:* 1) start with existing NLP datasets (e.g., QA, summarization, classification); 2) apply data cleaning and deduplication operators to remove noise and redundancy; 3) use template-based conversion operators; 4) apply quality filtering operators to discard anomalous samples; and 5) the pipeline outputs a large-scale, structurally SFT dataset<sup>[109, 162]</sup>.

- *Human/Crowdsourced Annotation Workflow:* 1) begin with task descriptions and manually designed examples; 2) engage human annotators or crowdworkers to create candidate instruction-response pairs; 3) apply consistency checking and expert adjudication operators to ensure reliability; 4) use formatting and deduplication operators to standardize outputs into the SFT format; and 5) the pipeline outputs a small but high-quality SFT dataset<sup>[113, 114, 174]</sup>.

- *LLM-Synthetic Workflow:* 1) construct a set of seed instructions from human input or existing corpora; 2) employ LLMs to generate instruction-response pairs; 3) apply automatic quality filtering operators to eliminate low-quality outputs; 4) perform human spot-checking as final validation; and 5) the pipeline outputs a large and diverse SFT dataset<sup>[175–178]</sup>.

- *Dialogue and Process-Supervision Workflow*: 1) begin with multi-turn user-assistant interaction logs; 2) apply conversation segmentation operators to preserve context and label speaker roles; 3) use structuring operators to convert into instruction-response format; 4) for reasoning-intensive tasks, insert process supervision operators; 5) apply privacy redaction and safety filtering operators to remove sensitive or unsafe content; and 6) the pipeline outputs a dataset tailored for conversational and reasoning capabilities<sup>[112, 152, 179–181]</sup>.

To better systematize operators and workflows, recent studies have introduced the concept of data preparation systems. An example is our DataFlow framework<sup>[173]</sup>, which abstracts core SFT operators into composable modules and supports their automated execution. Compared with single workflows, DataFlow emphasizes modularity, flexibility, and reproducibility, enabling researchers to efficiently build and optimize SFT pipelines within a unified platform.

## 4.2 Reinforcement Learning

Reinforcement learning (RL) data is typically constructed based on algorithms such as Group Relative Policy Optimization (GRPO)<sup>[182, 183]</sup>, where rewards are computed from model outputs. The data collection process often resembles that of SFT, but in many cases only the final answers are required to determine the reward. Consequently, this subsection focuses primarily on the filtering strategies applied to RL data.

A central challenge in RL for LLM reasoning is that many prompts are either too easy (nearly always solved) or too hard (almost never solved), yielding little or no learning signal under policy-gradient style objectives. This motivates data filtering strategies that preferentially train on examples with informative gradients. We summarize offline and online difficulty-aware filters.

*Difficulty-Aware Offline Filtering*. Before RL run, one can estimate a prompt-level pass rate by sampling  $k$  rollouts from the current (or reference) policy and verifying correctness with a task-specific checker. Prompts with intermediate pass rates (e.g.,  $\hat{p} \in [0.2, 0.8]$ ) are retained, while very-easy or very-hard items are downweighted or removed. This mid-difficulty curation is now common in reasoning-focused RL with verifiable rewards (RLVR) because

mid-band pass rates maximize the chance of receiving differentiated rewards across rollouts, hence informative gradients<sup>[184]</sup>.

*Difficulty-Aware Online Filtering*. Under GRPO<sup>[182, 183]</sup>, difficulty can be tracked during training by computing the on-the-fly pass rate per prompt from the group of rollouts used to form GRPO’s group-relative advantage. A recent study provides theoretical and empirical support for balanced online difficulty filtering, i.e., retaining prompts whose current pass rate stays near the middle band and replacing the rest with alternatives, thus keeping batch difficulty centered around the model’s evolving competence. This improves both sample and time efficiency, delivering up to 10 percentage points on American Invitational Mathematics Examination (AIME) over plain GRPO while needing fewer steps to surpass the baseline’s best reward<sup>[185]</sup>. In practice, asynchronous sampling is used to maintain fixed batch size while replacing filtered-out prompts<sup>[185]</sup>.

Effective RL with GRPO relies on carefully filtered data and reliable reward signals. A central strategy is to use verifiable rewards and enforce strict output formats (e.g., boxed final answers, tags, or compile/run checks) so that correctness remains unambiguous and filterable<sup>[183]</sup>. GRPO’s groupwise sampling further enables online estimation of per-prompt pass rates by targeting a mid-band difficulty level—the zone of proximal development—while refreshing the prompt pool as the model improves, thereby maintaining strong learning signals<sup>[185]</sup>. Together, these practices ensure that compute is concentrated on examples with high information value, yielding reproducible gains in reasoning RL.

## 4.3 Reinforcement Learning from Human Feedback

Reinforcement learning from human feedback (RLHF) has become a cornerstone technique for aligning LLMs with human preferences. In contrast to SFT, which directly learns from instruction-response pairs, and pure RL, which uses a golden answer to verify rewards, RLHF leverages human preference data to train a reward model (RM), which subsequently guides the optimization of the LLM through reinforcement learning. The preparation of RLHF data is therefore critical, as it directly impacts the reliability, stability, and generalization of the resulting model<sup>[186–188]</sup>.

### 4.3.1 Data Collection

The first step in RLHF is the collection of human preference data. A common paradigm is to generate multiple candidate responses for a given prompt and then ask human annotators to provide preferences through pairwise comparisons or rankings<sup>[186]</sup>. For example, InstructGPT<sup>[112]</sup> utilizes pairwise annotations to construct a large-scale preference dataset. Beyond binary choices, recent work explores natural language feedbacks such as critiques and revisions, which provide richer signals and have shown higher data efficiency<sup>[187]</sup>.

Data sources for RLHF include: 1) synthetic responses generated by SFT models or diverse model checkpoints, which ensure response variety; 2) crowd-sourced preference annotations, often with domain-specific guidelines to ensure consistency; and 3) hybrid approaches that combine limited human feedback with auxiliary signals such as proxy models or heuristics to reduce labeling costs<sup>[188]</sup>. Ensuring domain diversity and balanced coverage across different prompt categories is essential for avoiding bias in reward model training.

### 4.3.2 Data Filter

After collecting preference data, an essential stage in RLHF is data filtering, which aims to eliminate low-quality or noisy samples and improve the signal-to-noise ratio of the training corpus. Typical strategies include checking inter-annotator consistency and discarding samples with low agreement, calibrating annotators with gold-standard examples to reduce bias, and removing toxic, factually incorrect, or biased responses that compromise alignment. In addition, normalization and de-duplication ensure consistent annotation formats, while margin-based filtering discards pairs with negligible preference differences or low annotator confidence. More advanced methods compare annotated responses with model outputs and remove samples that are not clearly superior<sup>[189]</sup>. Recent work also emphasizes targeted annotation of hard cases where the model is most likely to fail, thereby maximizing the utility of limited annotation budgets<sup>[190]</sup>. Collectively, these approaches highlight that careful data filtering is critical for constructing robust and reliable reward models in RLHF.

### 4.3.3 Data Evaluation

Evaluating RLHF data is critical for ensuring that

reward models generalize and align with human intent. Beyond basic inter-annotator agreement<sup>[186]</sup>, recent work emphasizes adjusting training with data quality estimates<sup>[191]</sup>, validating reward models through held-out preference prediction<sup>[112]</sup>, and probing consistency across semantically similar prompts<sup>[192]</sup>. More advanced approaches incorporate hierarchical reward evaluation<sup>[193]</sup>, systematic benchmarks such as RewardBench<sup>[194]</sup>, and analyses of RM quality impacts<sup>[195]</sup>. Fine-grained methods like intra-trajectory consistency<sup>[196]</sup> and data-efficient strategies such as targeted human feedback<sup>[190]</sup> further highlight that robust evaluation requires both statistical reliability and downstream alignment testing.

### 4.3.4 RLHF Data Generation

Beyond human-annotated preferences, a growing line of work constructs synthetic RLHF data by replacing or supplementing human raters with AI feedback (RLAIF) and related judge-based signals. Constitutional AI<sup>[197]</sup> introduces critique-and-revision pipelines guided by a set of principles (e.g., harmlessness, helpfulness), yielding preference pairs and rewritten outputs without human labels for safety. RLAIF scales this idea: a strong LLM (or a committee of LLMs) is prompted as a judge to produce pairwise preferences or scalar rewards that train an RM or drive on-policy optimization; direct RLAIF (d-RLAIF) removes the RM entirely by querying the judge online during RL<sup>[198]</sup>. Self-rewarding approaches go further: the policy (or a sibling model) learns to produce its own rewards via LLM-as-a-judge prompting, improving both the policy and its internal judge through rounds of preference optimization<sup>[199]</sup>. Because LLM-as-a-judge introduces biases and variance, recent studies emphasize reliability techniques such as pairwise (rather than score-only) judging, rubric-grounded prompts, multi-judge aggregation, and meta-evaluation against human labels<sup>[200, 201]</sup>.

## 5 Datasets and Corresponding Workflows

In this section, we summarize commonly used datasets that are readily applicable for LLM research, together with the typical data preparation workflows associated with them. For better clarity, [Table 1](#) compares the contrasts of data preparation in the Pre-Training, Continual Pre-Training, and Post-Training stage.

**Table 1.** Comparison of Recurring Data Operations Across Different Training Stages

| Stage                  | Typical Data Scale        | Data Collection Cost       | Data Filtering        |
|------------------------|---------------------------|----------------------------|-----------------------|
| Pre-training           | $10^9$ – $10^{12}$ tokens | Low per-sample, high total | Heuristic & scalable  |
| Continual pre-training | $10^7$ – $10^9$ tokens    | Medium                     | Hybrid                |
| Post-training          | $10^3$ – $10^6$ samples   | High per-sample            | Fine-grained & strict |

### 5.1 Pre-Training and Continual Pre-Training Datasets: Overview and Practical Notes

Pre-training datasets are foundational for the initial training of LLMs, as well as for their continual refinement. The distinction between pre-training (PT) and continual pre-training (CPT) is crucial, as PT datasets aim to provide a broad, diverse knowledge base, while CPT datasets focus on keeping models up to date with emerging trends, new facts, and shifting language patterns. As summarized in Table 2, the datasets discussed here are essential for these training stages, offering both general knowledge and domain-specific expertise, while also ensuring that models can adapt to new challenges over time.

*General Web Pre-Training.* General web corpora provide the foundation for large-scale pre-training by offering broad and diverse coverage of knowledge. Early efforts such as C4<sup>[46]</sup> and The Pile<sup>[60]</sup> demonstrated the value of large curated web datasets: C4 focuses on English text derived from Common Crawl, while The Pile aggregates 22 carefully selected sources to balance diversity and quality. Building on these foundations, RedPajama-Data-1T<sup>⑫</sup> scales to over a trillion tokens, recreating the LLaMA-style mixture that spans web data, code, and academic content. More recent initiatives, including FineWeb<sup>[15]</sup> and Zyd-2<sup>[203]</sup>, further push scale and refinement, leveraging cross-source deduplication, model-based filtering, and advanced cleaning pipelines to deliver multi-trillion-token corpora. Across these generations, the common thread is the reliance on large-scale web crawls combined with increasingly sophisticated filtering and deduplication techniques<sup>[58, 63]</sup> to ensure both coverage and quality.

*Multilingual Pre-Training.* Multilingual corpora extend pre-training beyond English by covering diverse linguistic contexts, usually built from large-scale crawls and refined with language identification, cleaning, deduplication, and licensing checks. Early large-scale efforts such as ROOTS<sup>[26]</sup> set a benchmark with

59 languages, emphasizing ethical and legal considerations alongside rigorous filtering. Subsequent work scaled further: CulturaX<sup>[11]</sup> expands coverage to 167 languages with 6.3T tokens processed through advanced filtering pipelines, while HPLTv2<sup>[204]</sup> pushes toward even larger and more balanced multilingual corpora, addressing issues of underrepresented languages with improved quality control. Across these generations, the common trajectory is toward broader coverage and higher-quality cross-lingual representation, enabling models to achieve robust multilingual capabilities.

*Domain-Specific Pre-Training.* Beyond general and multilingual corpora, specialized datasets in technical domains play a key role in enabling models to reason about structured symbolic content. Early large-scale efforts such as The Stack v2<sup>[205]</sup> drew from Software Heritage, offering 67.5 TB of raw code reduced to about 900B tokens after filtering and deduplication, covering a wide range of programming languages together with documentation and development artifacts. More recently, OpenWebMath<sup>[18]</sup> targets math-heavy web content, applying math-aware filtering to curate 14.7B tokens of equations and explanations, tailored for mathematical reasoning tasks. Together, these corpora illustrate a trajectory from broad code collections to more focused datasets for symbolic domains like mathematics, reflecting a broader shift toward training models capable of handling programming, algorithmic reasoning, and problem-solving in technical contexts.

*Multilingual Continual Pre-Training.* Multilingual continual pre-training datasets ensure that models stay aligned with emerging knowledge, evolving language patterns, and shifting global contexts across diverse languages. EMMA-500 (MaLA corpus)<sup>[81]</sup> contains 74B tokens spanning 546 languages with extensive cleaning and deduplication, enabling multilingual models to remain updated. Building on this, AU-RORA-M<sup>[83]</sup> have 435B tokens of multilingual and code data, introducing synthetic rephrasing and human-reviewed safety red-teaming for a controlled two-

<sup>⑫</sup><https://github.com/togethercomputer/RedPajama-Data/>, Jan. 2026.

**Table 2.** Representative PT and CPT Datasets with Their Data Preparation Workflows Across Domains (Simplified)

| Dataset (Link)                                       | Domain/Task               | Data Workflow Detail                                                                                                                                                                               | Size                              | Time      | Stage |
|------------------------------------------------------|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------|-----------|-------|
| <a href="#">FineWeb</a>                              | General/Web PT            | Crawl CC → text extraction → language detect → quality filter → dedup → PII handling                                                                                                               | About 18.5T tokens (v1.4)         | 2024      | PT    |
| <a href="#">RefinedWeb</a>                           | General/Web PT            | CC only; heavy cleaning and dedup; public research extract                                                                                                                                         | 600B tokens                       | 2023      | PT    |
| <a href="#">RedPajama-Data-1T</a>                    | General/LLaMA A-style mix | Recreate LLaMA mix (web, code, arXiv, Wiki, SE) → clean → dedup                                                                                                                                    | About 1.2T tokens                 | 2023      | PT    |
| <a href="#">The Pile</a>                             | General/English           | 22 curated sources → cleaning → dedup → mixture spec                                                                                                                                               | About 825 GiB                     | 2020      | PT    |
| <a href="#">C4</a>                                   | General/Web               | From CC → English detection → heuristic cleaning → dedup                                                                                                                                           | About 807 GiB (c4/en)             | 2019      | PT    |
| <a href="#">ROOTS (BigScience)</a>                   | Multilingual/Web+         | 59 languages; ethics/licensing review → cleaning → dedup                                                                                                                                           | 1.6 TB                            | 2022      | PT    |
| <a href="#">CulturaX</a>                             | Multilingual/Web          | 167 languages; language ID → filters → dedup → per-language release                                                                                                                                | 6.3T tokens                       | 2023      | PT    |
| <a href="#">HPLT v2</a>                              | Multilingual/Web PT       | CC + Internet archive → quality filters → dedup → CC0 release                                                                                                                                      | About 8T tokens; 193 languages    | 2025      | PT    |
| <a href="#">Dolma</a>                                | Mixed                     | Combine multiple sources → clean → dedup → safety filters                                                                                                                                          | 3T tokens                         | 2023      | PT    |
| <a href="#">The Stack v2</a>                         | Code/pre-training         | From software heritage → license/language detect → filter + dedup → repository splits                                                                                                              | 67.5 TB raw/About 900B tokens     | 2024      | PT    |
| <a href="#">OpenWebMath</a>                          | Math/Web PT               | Identify math-heavy pages → clean boilerplate → math-aware filter → dedup                                                                                                                          | 14.7B tokens                      | 2023      | PT    |
| <a href="#">EMMA-500 (MaLA corpus)</a>               | Multilingual              | Compile & curate MaLA across 546 languages → heavy cleaning/dedup → continual pre-training of Llama-2-7B; corpus & weights released                                                                | 546 languages (> 74B tokens)      | 2024–2025 | CPT   |
| <a href="#">AURORA-M</a>                             | Multilingual + Code       | Two-stage CPT: CAP on multilingual web/code (processed) + CAT alignment mix; human-reviewed safety red-teaming                                                                                     | 435B tokens                       | 2024      | CPT   |
| <a href="#">Nemotron-CC (v2)</a>                     | General/Web               | Multi-CC snapshots (2024–2025) → English filter → synthetic rephrasing (LLM) → global dedup; includes Diverse QA (15-language translations)                                                        | About 6.586T tokens               | 2025      | PT    |
| <a href="#">Nemotron-MIND</a>                        | Math                      | Convert OpenWebMath math text → multi-style synthetic dialogues (7 styles) → heuristic filtering → CPT 7B models                                                                                   | > 138B tokens                     | 2025      | CPT   |
| <a href="#">YuLan-Mini Datasets</a>                  | Code/Math/Science         | Open pre-training resources (cleaning, schedule/annealing, curated + synthetic math/code) with per-phase composition detailed                                                                      | 1.08T tokens; HF release: subsets | 2024–2025 | PT    |
| <a href="#">MegaMath-Web-Pro-Max<sup>[202]</sup></a> | Math/Web                  | Common Crawl math-heavy pages → math/quality scoring (math_score, finemath_score, lang_score) → language detect & filters → global dedup → structured Parquet with cc-path, domain, timestamp, URL | 69.2 million (M) rows             | 2025      | CPT   |

Note: The electronic version of the paper provides jump links to the websites of each dataset. In “Domain/Task”, if the “/” symbol is absent, the term should be interpreted as referring to Domain; dedup: deduplication.

stage process. At larger scale, Nemotron-CC v2<sup>[86]</sup> is constructed from multi-snapshot Common Crawl with LLM-based rephrasing, producing 6.6T tokens to refresh global web content. Nemotron-MIND<sup>[105]</sup> further extends CPT to mathematical reasoning, transforming OpenWebMath into 138B tokens of multi-style synthetic dialogues. Collectively, these corpora illustrate a trajectory from multilingual coverage to diverse, safety-aware, and large-scale updates across both general and specialized domains.

*Domain-Specific Continual Pre-Training.* Beyond

multilingual updates, domain-specific CPT datasets sustain expertise in rapidly evolving technical areas. YuLan-Mini<sup>[101]</sup> provides 1.08T tokens across math, code, and science, combining curated and synthetic data for continual domain adaptation. MegaMath-Web-Pro-Max<sup>[20]</sup> focuses narrowly on math, extracting math-heavy content from Common Crawl and applying domain-specific filtering and quality scoring to refine reasoning capabilities. These efforts demonstrate a shift from broad multilingual corpora toward targeted resources that maintain high precision in

specialized fields such as science, math, and programming.

*Workflow Patterns and Cautions.* Across pre-training and continual pre-training corpora, we observe recurring steps: large-scale crawling or standardized mixtures for reproducibility, filtering against harmful content, social bias, redundancy and contamination, quality and format issues, enrichment with domain or multilingual data, synthetic augmentation, and structured reformatting. For evaluation, it is important to emphasize contamination checks, freshness validation, and targeted benchmarks to ensure sustained relevance. Practically, we recommend beginning with one large-scale general corpus (e.g., FineWeb or RedPajama), one reproducible mixture (e.g., C4 or HPLTv2), and one specialized dataset (e.g., OpenWebMath or The Stack v2), then complementing with continual refreshes such as Nemotron-CC v2 or AURORA-M.

## 5.2 Post-Training Datasets: Overview and Practical Notes

Post-training data for alignment and RLHF commonly falls into the following four buckets: 1) general instruction and chat corpora, 2) domain-specialized reasoning sets (e.g., math, science), 3) code datasets with verifiable signals (e.g., tests, executors), and 4) tool-use/web-agent datasets. Below, we highlight widely used, easy-to-start datasets corresponding to Table 3 and summarize why they are practical.

*General Instruction and Chat.* Infinity-Instruct provides a two-phase pipeline that first selects high-quality instructions from an over 100M pool (hybrid curation) and then synthesizes chat-style data via instruction evolution and diagnostic filtering, covering both foundational and multi-turn capabilities<sup>[206]</sup>. OpenHermes 2.5 aggregates 1M open instruction and chat samples (largely synthetic) with a focus on breadth and cleanliness, and is frequently used to quickly bootstrap SFT for assistant-style behavior<sup>[13]</sup>. ShareGPT52K traces back to user-shared, multi-turn conversations; typical curation includes markup normalization, safety filtering, and deduplication, and its usage was popularized by Vicuna-style instruction tuning<sup>[207]</sup>.

*Math Instruction and Reasoning.* OpenMathInstruct-2 (14M pairs) distills from strong teachers (e.g.,

Llama-3.1-405B) with explicit length or format control; it is effective for SFT and offers clear scaling behavior<sup>[19]</sup>. NuminaMath-CoT unifies CoT templates over competition-level problems at scale (0.86M), making it a strong resource for training and analysis of mathematical reasoning<sup>[208, 209]</sup>. For challenging evaluation, Omni-MATH (4 428 olympiad-level problems) introduces a broad taxonomy and difficulty ladder that stresses high-level reasoning beyond GSM8K/MATH<sup>[210]</sup>.

*General Reasoning (Cross-Domain).* SYNTHETIC-1 releases a large synthetic, verifier-backed reasoning set across math, code, and logic; it is useful as a high-signal complement for post-training, with LLM or symbolic verifiers enhancing filter quality<sup>[14]</sup>.

*Code Pre-Training and SFT with Verifiability.* The Stack v2 underpins code pre-training (over 3B files, over 600 languages) with licensing and deduplication heuristics, and is the base for many modern code LLMs<sup>[205]</sup>. At the SFT level, EpiCoder-func-380k synthesizes function-level tasks guided by feature mining and complexity control (Python), enabling fine-grained capability shaping<sup>[211]</sup>. KodCode-V1 contributes diverse, challenging tasks with unit tests and verified solutions across multiple sub-domains, which is particularly helpful for RL/SFT with automatic rejection sampling<sup>[212]</sup>.

*Software-Engineering Evaluation.* SWE-bench Verified provides a human-validated subset of real GitHub issues, strengthening reliability over the original SWE-bench; it remains a de facto target for coding agents, though recent work warns about contamination/memorization and calls for robust protocols<sup>[213–215]</sup>.

*Science and Tool-Use/Web Agents.* ScienceQA (NeurIPS 2022) offered multimodal K-12 questions with explanations/rationales, supporting SFT and rationale-aware evaluation<sup>[216]</sup>. ToolBench (ToolLLM) scales tool-use instruction data over more than 16 000 real APIs, automating instruction construction and denoising to train tool-invocation competence<sup>[217]</sup>. WebWalkerQA evaluates web traversal and deep information seeking across multi-hop site navigation, useful for training and testing web agents<sup>[218]</sup>.

*Common Workflow Patterns and Cautions.* Across these corpora, we observe recurring workflow stages: data selection (hybrid heuristics and embedding-based retrieval), synthetic generation with strong teacher models, structured formatting (e.g., role tags, length

<sup>[13]</sup><https://huggingface.co/datasets/teknium/OpenHermes-2.5>, Jan. 2026.

<sup>[14]</sup><https://www.primeintellect.ai/blog/synthetic-1-release>, Jan. 2026.

**Table 3.** Representative Post-Training Datasets and Their Data Preparation Workflows Across Different Domains

| Dataset (Link)                              | Domain/Task                 | Data Workflow Detail                                                                                                                                                                          | Size                                 | Time | Stage           |
|---------------------------------------------|-----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------|------|-----------------|
| <a href="#">Infinity-Instruct</a>           | General/instruction & chat  | Phase-1: curate foundational instructions from > 100M pool via hybrid selection & dedup → Phase-2: synthesize chat instructions via instruction evolution → diagnostic filtering → formatting | 7.4M (foundational) + 1.5M (chat)    | 2025 | SFT             |
| <a href="#">OpenHermes 2.5</a>              | General/instruction & chat  | Compile open instruction/chat sets + GPT-4-distilled samples → cleaning/curation → dedup → ShareGPT-style multi-turn formatting                                                               | About 1.0M samples                   | 2024 | SFT             |
| <a href="#">ShareGPT</a>                    | General/dialogue            | User-shared ChatGPT conversations from ShareGPT.com → HTML-to-Markdown conversion → cleaning & safety filtering → dedup → multi-turn formatting                                               | About 90 000 (90k) conversations     | 2023 | SFT/RLHF        |
| <a href="#">OpenMathInstruct-2</a>          | Math/instruction            | Seed from GSM8K/MATH → Llama-3.1-405B teacher generation → format/length control → light filtering                                                                                            | 14M pairs                            | 2024 | SFT             |
| <a href="#">NuminaMath-CoT</a>              | Math/reasoning              | Competition problems scraping → CoT templating → normalization → filtering                                                                                                                    | About 860k pairs                     | 2024 | SFT             |
| <a href="#">Omni-MATH</a>                   | Math/benchmark              | Olympiad problems curated → taxonomy/difficulty labels → formatting                                                                                                                           | 4 428 problems                       | 2024 | Evaluation      |
| <a href="#">SYNTHETIC-1 (Math and Code)</a> | General reasoning           | DeepSeek-R1 generation → crowdsourced compute → verifier checks (LLM/symbolic) → filtered splits                                                                                              | ≥ 1.4M tasks                         | 2025 | SFT (reasoning) |
| <a href="#">The Stack v2</a>                | Code/pre-training           | SWHIDs crawl → license detection → dedup/heuristic filters → repository-grouped training split                                                                                                | 67.5 TB raw/About 900B tokens        | 2024 | PT              |
| <a href="#">EpiCoder-func-380k</a>          | Code/function-level         | Instruction synthesis → function-level code (Python) → diversity/complexity curation → cleaning                                                                                               | 380k samples                         | 2025 | SFT             |
| <a href="#">KodCode-V1</a>                  | Code/instruction & RL       | Synthetic tasks → verifiable solutions+unit tests → auto-evaluation → dedup → multi-style (Instruct/Complete/OJ)                                                                              | 487k items                           | 2025 | SFT/RLHF        |
| <a href="#">SWE-bench Verified</a>          | Code/software Eng.          | GitHub issues → human-validated solvable subset → unit-test verification protocol                                                                                                             | 500 issues                           | 2024 | Evaluation      |
| <a href="#">ScienceQA</a>                   | Science QA/multimodal MCQ   | K-12 science questions collection → image-text context curation → rationale/explanation annotation (usable as CoT) → formatting                                                               | 21 208 questions                     | 2022 | Evaluation/SFT  |
| <a href="#">ToolBench (ToolLLM)</a>         | Tool use/instruction tuning | Crawl 16 464 REST APIs (RapidAPI) → LLM-generated single/multi-tool instructions & solutions → automatic evaluation & denoising → instruction-style formatting                                | 16k+ APIs (large instruction corpus) | 2023 | SFT (tool-use)  |
| <a href="#">WebWalkerQA</a>                 | Tool use/web traversal QA   | Curate deep multi-hop site tasks → author QAs with gold navigation paths → bilingual (EN/ZH) formatting → test split                                                                          | 680 QAs                              | 2025 | Evaluation      |

Note: In “Domain/Task”, if the “/” symbol is absent, the term should be interpreted as referring to Domain; dedup: deduplication, EN: English, ZH: Chinese.

or step control, schema normalization), safety filtering, and deduplication. For code and tool-use data, additional verifiable signals—such as unit tests, executors, and API schemas—are commonly employed to ensure correctness.

Importantly, these workflows differ substantially in scale and emphasis across training stages. While pre-training pipelines typically prioritize throughput and operate on extremely large corpora, post-training

data preparation is usually conducted at much smaller scales (often ranging from thousands to millions of samples) and places greater emphasis on fine-grained quality control, verification, and task alignment. This makes post-training workflows particularly suitable for small research teams, low-resource domains, and resource-constrained settings.

For software engineering-style evaluation, we recommend contamination checks, stronger test-set aug-

mentation, and human validation when feasible<sup>[214, 215]</sup>. In practice, a lightweight yet effective setup is to start with one general assistant dataset (e.g., Infinity-Instruct or OpenHermes), one domain-specialized dataset (e.g., OpenMathInstruct-2 or NuminaMath-CoT), and one verifiable code dataset (e.g., EpiCoder or KodCode), and then introduce targeted benchmarks (e.g., Omni-MATH, SWE-bench Verified, ScienceQA) to iteratively refine data quality.

### 5.3 Comparative Analysis of Workflow Strategies

The impact of data preparation on downstream model performance depends on task characteristics, model capacity, and training stage. Despite this variability, empirical studies consistently show that rule-based and model-based data preparation strategies differ in efficiency, scalability, and quality control. As a result, effective large-scale workflows are rarely monolithic. Instead, they are best viewed as compositional systems that combine complementary mechanisms through a coarse-to-fine processing paradigm, adapted to different stages of model training. A central consideration in data preparation is the trade-off between efficiency and quality.

*Rule-Based Methods.* These methods are distinguished by low computational cost and strong scalability, enabling their application to pre-training corpora with trillions of tokens. In practice, they serve as a foundational sanitation layer by removing duplicate content, boilerplate text, and surface-level noise. Their contribution to model performance is primarily preventative, reducing risks such as memorization and benchmark contamination. However, because these methods rely on shallow lexical signals, they have limited ability to improve semantic diversity or informational density.

*Model-Based Methods.* These approaches offer finer-grained quality control by leveraging learned representations or inference from trained models. Techniques such as semantic deduplication and model-driven scoring can identify semantic redundancy, logical inconsistencies, and low-value samples that rule-based heuristics often miss. These methods frequently improve training efficiency by increasing the effective value per token. Their main limitation is computational cost, which restricts their direct application to raw web-scale corpora and motivates their use on smaller or higher-value data subsets.

The relative importance of these strategies varies across training stages. During pre-training, data scale is the dominant constraint, and workflows prioritize coverage while maintaining basic cleanliness. Efficient rule-based filtering therefore plays a central role, with model-based components used sparingly. In continual pre-training and supervised fine-tuning, the focus shifts toward value per token, as data volumes decrease and precision becomes critical. At these stages, model-based operators are increasingly employed to support domain adaptation and instruction alignment.

Recent data pipelines demonstrate that the strongest performance is achieved by integrating rule-based and model-based strategies within a unified workflow. Typically, inexpensive rule-based filters remove most low-quality data, after which model-based methods are applied to the remaining candidates. This coarse-to-fine structure balances scalability and effectiveness, concentrates computational resources on high-impact data, and aligns naturally with the staged process of LLM training.

*Empirical Takeaways (Impact vs Cost).* As a survey, we synthesize empirical evidence reported in prior work rather than introduce new experiments. Across the literature, the data components with the largest impact on downstream performance are typically 1) high-quality data synthesis (especially when guided by strong teachers or verifiable signals) and 2) principled data cleaning (e.g., filtering, deduplication, and contamination removal), which consistently improves generalization. In contrast, formatting and normalization usually provide more incremental gains by improving training stability and efficiency. From a cost perspective, selection and lightweight cleaning are relatively inexpensive at scale, whereas synthesis and verification are the most compute-intensive. This cost-benefit trade-off explains why post-training commonly favors smaller data volumes with higher per-sample quality, while pre-training emphasizes throughput at scale.

## 6 Future Directions

### 6.1 Data Preparation Systems

The future of data preparation for LLMs lies in building more systematic, automated, and scalable pipelines that can continuously adapt to the growing complexity of multimodal and multilingual data. Current approaches are often fragmented, relying on ad hoc heuristics or task-specific curation. A promising

direction is the design of unified data preparation systems that integrate data acquisition, cleaning, filtering, augmentation, and evaluation into a cohesive framework<sup>[173]</sup>. Such systems should be modular, allowing researchers to flexibly plug in rule-based methods, statistical techniques, or LLM-powered agents depending on the target application. Moreover, future systems will need to emphasize efficiency and reproducibility, supporting large-scale yet transparent data management that ensures traceability of training corpora and facilitates responsible AI development.

## 6.2 Next Generation of Data Management

A crucial future direction is to advance from traditional data management practices toward more fine-grained and semantically meaningful organization of training corpora. Unlike conventional database-style management, which focuses primarily on storage and access efficiency, data preparation for LLMs demands multi-dimensional quality characterization. For example, each dataset should be annotated with quality labels that indicate not only its domain and scale, but also its strengths and weaknesses across dimensions such as factual accuracy, linguistic diversity, safety, and alignment with human instructions. While platforms like Hugging Face have enabled unprecedented accessibility to a vast collection of datasets, the lack of standardized quality-oriented metadata limits their effective reuse. Future work should therefore explore principled frameworks for dataset documentation, benchmarking, and labeling, so that researchers can better match data resources to intended applications. This richer perspective on data management will play a critical role in ensuring that LLMs are trained on data that is not only abundant, but also trustworthy, transparent, and purpose-driven.

## 7 Conclusions

In this survey, we provided a comprehensive overview of data preparation for large language models (LLMs), covering the full pipeline from pre-training to continual pre-training and post-training. We introduced a novel taxonomy that begins from data operators as fundamental processing units and extends to data workflows as their compositions. This perspective not only highlights the modularity and flexibility of data preparation but also clarifies how different operators and workflows can be tailored to

the unique demands of LLM training.

Beyond theoretical categorization, we also compiled and organized widely used datasets together with their associated preparation workflows. This serves as a practical reference for both beginners entering the field and experienced practitioners seeking to design or improve data-centric pipelines for LLMs. By combining methodological summaries with hands-on resources, this survey lowers the entry barrier to data preparation while also providing a structured understanding of the state of the art.

Looking forward, we believe that data preparation will continue to play a pivotal role in the evolution of LLMs. Challenges such as dynamic data quality assessment, workflow automation, multi-modal integration, and scalable data management remain largely open. Addressing these issues requires not only more sophisticated operators and workflows but also systematic evaluation frameworks and community-shared resources. We hope this survey can serve as both a roadmap and a catalyst for future research on advancing data preparation for LLMs.

**Conflict of Interest** Bin Cui is an editorial board member for the Journal of Computer Science and Technology and was not involved in the editorial review of this article. The authors declare that there are no other competing interests.

## References

- [1] Bai T, Liang H, Wan B, Xu Y, Li X, Li S, Yang L, Li B, Wang Y, Cui B, Huang P, Shan J, He C, Yuan B, Zhang W. A survey of multimodal large language model from a data-centric perspective. arXiv: 2405.16640, 2024. <https://arxiv.org/abs/2405.16640>, Jan. 2026.
- [2] Zha D, Bhat Z P, Lai K H, Yang F, Jiang Z, Zhong S, Hu X. Data-centric artificial intelligence: A survey. *ACM Computing Surveys*, 2025, 57(5): 129. DOI: [10.1145/3711118](https://doi.org/10.1145/3711118).
- [3] Zhou X, He J, Zhou W, Chen H, Tang Z, Zhao H, Tong X, Li G, Chen Y, Zhou J, Sun Z, Hui B, Wang S, He C, Liu Z, Zhou J, Wu F. A survey of LLM  $\times$  DATA. arXiv: 2505.18458, 2025. <https://arxiv.org/abs/2505.18458>, Jan. 2026.
- [4] Wang Z, Zhong W, Wang Y, Zhu Q, Mi F, Wang B, Shang L, Jiang X, Liu Q. Data management for training large language models: A survey. arXiv: 2312.01700, 2023. <https://arxiv.org/abs/2312.01700>, Jan. 2026.
- [5] Liu Y, Cao J, Liu C, Ding K, Jin L. Datasets for large language models: A comprehensive survey. *Artificial Intelligence Review*, 2025, 58(12): 403. DOI: [10.1007/s10462-025-11403-7](https://doi.org/10.1007/s10462-025-11403-7).

- [6] Selvam P, Patel H, Surendran S, Singh S. Data preparation for fine tuning large language models. In *Proc. the 8th International Conference on Data Science and Management of Data*, Dec. 2024, pp.378–380. DOI: [10.1145/3703323.3704802](https://doi.org/10.1145/3703323.3704802).
- [7] Wenzek G, Lachaux M A, Conneau A, Chaudhary V, Guzmán F, Joulin A, Grave E. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proc. the 12th Language Resources and Evaluation Conference*, May 2020, pp.4003–4012.
- [8] Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I. Language models are unsupervised multitask learners. *OpenAI Blog*, 2019, 1(8): Article No. 9. [https://cdn.openai.com/better-language-models/language\\_models\\_are\\_unsupervised\\_multitask\\_learners.pdf](https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf), Jan. 2026.
- [9] 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. *Journal of Machine Learning Research*, 2020, 21(140): 1–67.
- [10] Overwijk A, Xiong C, Liu X, VandenBerg C, Callan J. ClueWeb22: 10 billion web documents with visual and semantic information. arXiv: 2211.15848, 2022. <https://arxiv.org/abs/2211.15848>, Jan. 2026.
- [11] Nguyen T, Van Nguyen C, Lai V D, Man H, Ngo N T, Dernoncourt F, Rossi R A, Nguyen T H. CulturaX: A cleaned, enormous, and multilingual dataset for large language models in 167 languages. In *Proc. the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation*, May 2024, pp.4226–4237.
- [12] Kargaran A H, Yvon F, Schutze H. GlotCC: An open broad-coverage CommonCrawl corpus and pipeline for minority languages. In *Proc. the 38th International Conference on Neural Information Processing Systems*, Dec. 2024, Article No. 540. DOI: [10.5555/3737916.3738456](https://doi.org/10.5555/3737916.3738456).
- [13] Weber M, Fu D Y, Anthony Q et al. RedPajama: An open dataset for training large language models. In *Proc. the 38th International Conference on Neural Information Processing Systems Track on Datasets and Benchmarks*, Dec. 2024. <https://openreview.net/pdf?id=lnuXaRpwwv>, Jan. 2026.
- [14] Soldaini L, Kinney R, Bhagia A et al. Dolma: An open corpus of three trillion tokens for language model pre-training research. In *Proc. the 62nd Annual Meeting of the Association for Computational Linguistics*, Aug. 2024, pp.15725–15788. DOI: [10.18653/v1/2024.acl-long.840](https://doi.org/10.18653/v1/2024.acl-long.840).
- [15] Penedo G, Kydliack H, Ben Allal L, Lozhkov A, Mitchell M, Raffel C, Von Werra L, Wolf T. The FineWeb datasets: Decanting the web for the finest text data at scale. In *Proc. the 38th International Conference on Neural Information Processing Systems*, Dec. 2024, Article No. 970. DOI: [10.5555/3737916.3738886](https://doi.org/10.5555/3737916.3738886).
- [16] Langlais P C, Rosas Hinostroza C, Nee M, Arnett C, Chizhov P, Jones E K, Girard I, Mach D, Stasenko A, Yamshchikov I P. Common corpus: The largest collection of ethical data for LLM pre-training. arXiv: 2506.01732, 2025. <https://arxiv.org/abs/2506.01732>, Jan. 2026.
- [17] Li J, Fang A, Smyrnis G et al. DataComp-LM: In search of the next generation of training sets for language models. In *Proc. the 38th International Conference on Neural Information Processing Systems*, Dec. 2024, Article No. 455. DOI: [10.5555/3737916.3738371](https://doi.org/10.5555/3737916.3738371).
- [18] Paster K, Dos Santos M, Azerbayev Z, Ba J. OpenWebMath: An open dataset of high-quality mathematical web text. In *Proc. the 12th International Conference on Learning Representations*, May 2024.
- [19] Toshniwal S, Du W, Moshkov I, Kisacanin B, Ayrapetyan A, Gitman I. OpenMathInstruct-2: Accelerating AI for math with massive open-source instruction data. In *Proc. the 13th International Conference on Learning Representations*, Apr. 2025.
- [20] Zhou F, Wang Z, Ranjan N, Cheng Z, Tang L, He G, Liu Z, Xing E P. MegaMath: Pushing the limits of open math corpora. arXiv: 2504.02807, 2025. <https://arxiv.org/abs/2504.02807>, Jan. 2026.
- [21] Wu C, Lin W, Zhang X, Zhang Y, Xie W, Wang Y. PMC-LLaMA: Toward building open-source language models for medicine. *Journal of the American Medical Informatics Association*, 2024, 31(9): 1833–1843. DOI: [10.1093/jamia/ocae045](https://doi.org/10.1093/jamia/ocae045).
- [22] Xie Q, Chen Q, Chen A et al. Me-LLaMA: Foundation large language models for medical applications. *Research Square*, 2024. DOI: [10.21203/rs.3.rs-4240043/v1](https://doi.org/10.21203/rs.3.rs-4240043/v1).
- [23] Touchent R, Godey N, Clergerie E. Biomed-Enriched: A biomedical dataset enriched with LLMs for pretraining and extracting rare and hidden content. arXiv: 2506.20331, 2025. <https://arxiv.org/pdf/2506.20331>, Jan. 2026.
- [24] Wang B, Xu C, Zhao X, Ouyang L, Wu F, Zhao Z, Xu R, Liu K, Qu Y, Shang F, Zhang B, Wei L, Sui Z, Li W, Shi B, Qiao Y, Lin D, He C. MinerU: An open-source solution for precise document content extraction. arXiv: 2409.18839, 2024. <https://arxiv.org/abs/2409.18839>, Jan. 2026.
- [25] Subramani N, Luccioni S, Dodge J, Mitchell M. Detecting personal information in training corpora: An analysis. In *Proc. the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*, Jul. 2023, pp.208–220. DOI: [10.18653/v1/2023.trustnlp-1.18](https://doi.org/10.18653/v1/2023.trustnlp-1.18).
- [26] Laurençon H, Saulnier L, Wang T et al. The BigScience ROOTS corpus: A 1.6TB composite multilingual dataset. In *Proc. the 36th International Conference on Neural Information Processing Systems*, Nov. 2022, Article No. 2306. DOI: [10.5555/3600270.3602576](https://doi.org/10.5555/3600270.3602576).
- [27] Qiu J, Lv H, Jin Z, Wang R, Ning W, Yu J, Zhang C B, Li Z, Chu P, Qu Y, Shi J, Lu L, Peng R, Zeng Z, Tang H, Lei Z, Hong J, Chen K, Fei Z, Xu R, Li W, Tu Z, Lin D, Qiao Y, Yan H, He C. WanJuan-CC: A safe and high-quality open-sourced English webtext dataset. arXiv:

- iv: 2402.19282, 2024. <https://arxiv.org/abs/2402.19282>, Jan. 2026.
- [28] Hathurusinghe R, Nejadgholi I, Bolic M. A privacy-preserving approach to extraction of personal information through automatic annotation and federated learning. In *Proc. the 3rd Workshop on Privacy in Natural Language Processing*, Jun. 2021, pp.36–45. DOI: [10.18653/v1/2021.privatenlp-1.5](https://doi.org/10.18653/v1/2021.privatenlp-1.5).
- [29] Gehman S, Gururangan S, Sap M, Choi Y, Smith N A. RealToxicityPrompts: Evaluating neural toxic degeneration in language models. In *Proc. the 2020 Findings of the Association for Computational Linguistics*, Nov. 2020, pp.3356–3369. DOI: [10.18653/v1/2020.findings-emnlp.301](https://doi.org/10.18653/v1/2020.findings-emnlp.301).
- [30] Welbl J, Glaese A, Uesato J, Dathathri S, Mellor J, Hendricks L A, Anderson K, Kohli P, Coppin B, Huang P S. Challenges in detoxifying language models. In *Proc. the 2021 Findings of the Association for Computational Linguistics*, Nov. 2021, pp.2447–2469. DOI: [10.18653/v1/2021.findings-emnlp.210](https://doi.org/10.18653/v1/2021.findings-emnlp.210).
- [31] Zeng W, Ren X, Su T *et al.* PanGu- $\alpha$ : Large-scale autoregressive pretrained Chinese language models with auto-parallel computation. arXiv: 2104.12369, 2021. <https://arxiv.org/abs/2104.12369>, Jan. 2026.
- [32] Xue L, Constant N, Roberts A, Kale M, Al-Rfou R, Siddhant A, Barua A, Raffel C. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proc. the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Jun. 2021, pp.483–498. DOI: [10.18653/v1/2021.naacl-main.41](https://doi.org/10.18653/v1/2021.naacl-main.41).
- [33] Penedo G, Malartic Q, Hesslow D, Cojocaru R, Alobaidli H, Cappelli A, Pannier B, Almazrouei E, Launay J. The RefinedWeb dataset for falcon LLM: Outperforming curated corpora with web data only. In *Proc. the 37th International Conference on Neural Information Processing Systems*, Dec. 2023, Article No. 3464. DOI: [10.5555/3666122.3669586](https://doi.org/10.5555/3666122.3669586).
- [34] Joulin A, Grave E, Bojanowski P, Mikolov T. Bag of tricks for efficient text classification. In *Proc. the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, Apr. 2017, pp.427–431.
- [35] Feng S, Park C Y, Liu Y, Tsvetkov Y. From pretraining data to language models to downstream tasks: Tracking the trails of political biases leading to unfair NLP models. In *Proc. the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Jul. 2023, pp.11737–11762. DOI: [10.18653/v1/2023.acl-long.656](https://doi.org/10.18653/v1/2023.acl-long.656).
- [36] Nadeem M, Bethke A, Reddy S. StereoSet: Measuring stereotypical bias in pretrained language models. In *Proc. the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Aug. 2021, pp.5356–5371. DOI: [10.18653/v1/2021.acl-long.416](https://doi.org/10.18653/v1/2021.acl-long.416).
- [37] Meade N, Poole-Daynan E, Reddy S. An empirical survey of the effectiveness of debiasing techniques for pre-trained language models. In *Proc. the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, May 2022, pp.1878–1898. DOI: [10.18653/v1/2022.acl-long.132](https://doi.org/10.18653/v1/2022.acl-long.132).
- [38] Garg S, Perot V, Limtiaco N, Taly A, Chi E H, Beutel A. Counterfactual fairness in text classification through robustness. In *Proc. the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, Jan. 2019, pp.219–226. DOI: [10.1145/3306618.3317950](https://doi.org/10.1145/3306618.3317950).
- [39] Dinan E, Fan A, Williams A, Urbanek J, Kiela D, Weston J. Queens are powerful too: Mitigating gender bias in dialogue generation. In *Proc. the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Nov. 2020, pp.8173–8188. DOI: [10.18653/v1/2020.emnlp-main.656](https://doi.org/10.18653/v1/2020.emnlp-main.656).
- [40] Barikeri S, Lauscher A, Vulić I, Glavaš G. RedditBias: A real-world resource for bias evaluation and debiasing of conversational language models. In *Proc. the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Aug. 2021, pp.1941–1955. DOI: [10.18653/v1/2021.acl-long.151](https://doi.org/10.18653/v1/2021.acl-long.151).
- [41] Lu K, Mardziel P, Wu F, Amancharla P, Datta A. Gender bias in neural natural language processing. In *Logic, Language, and Security: Essays Dedicated to Andre Scedrov on the Occasion of His 65th Birthday*, Nigam V, Kirigin T B, Talcott C, Guttman J, Kuznetsov S, Loo B T, Okada M (eds.), Springer, 2020, pp.189–202. DOI: [10.1007/978-3-030-62077-6\\_14](https://doi.org/10.1007/978-3-030-62077-6_14).
- [42] Maudslay R H, Gonen H, Cotterell R, Teufel S. It's all in the name: Mitigating gender bias with name-based counterfactual data substitution. In *Proc. the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Nov. 2019, pp.5267–5275. DOI: [10.18653/v1/D19-1530](https://doi.org/10.18653/v1/D19-1530).
- [43] Dwork C, Hardt M, Pitassi T, Reingold O, Zemel R. Fairness through awareness. In *Proc. the 3rd Innovations in Theoretical Computer Science Conference*, Jan. 2012, pp.214–226. DOI: [10.1145/2090236.2090255](https://doi.org/10.1145/2090236.2090255).
- [44] Zayed A, Parthasarathi P, Mordido G, Palangi H, Shabianian S, Chandar S. Deep learning on a healthy data diet: Finding important examples for fairness. In *Proc. the 37th AAAI Conference on Artificial Intelligence*, Jun. 2023, pp.14593–14601. DOI: [10.1609/aaai.v37i12.26706](https://doi.org/10.1609/aaai.v37i12.26706).
- [45] Ngo H, Raterink C, Aratjo J G M, Zhang I, Chen C, Morisot A, Frosst N. Mitigating harm in language models with conditional-likelihood filtration. arXiv: 2108.07790, 2021. <https://arxiv.org/abs/2108.07790>, Jan. 2026.
- [46] Dodge J, Sap M, Marasović A, Agnew W, Ilharco G, Groeneveld D, Mitchell M, Gardner M. Documenting

- large Webtext corpora: A case study on the colossal clean crawled corpus. In *Proc. the 2021 Conference on Empirical Methods in Natural Language Processing*, Nov. 2021, pp.1286–1305. DOI: [10.18653/v1/2021.emnlp-main.98](https://doi.org/10.18653/v1/2021.emnlp-main.98).
- [47] OpenAI. GPT-4 technical report. arXiv: 2303.08774, 2021. <https://arxiv.org/pdf/2303.08774>, Jan. 2026.
- [48] Brown T B, Mann B, Ryder N et al. Language models are few-shot learners. In *Proc. the 34th International Conference on Neural Information Processing Systems*, Dec. 2020, Article No. 159. DOI: [10.5555/3495724.3495883](https://doi.org/10.5555/3495724.3495883).
- [49] Chowdhery A, Narang S, Devlin J et al. PaLM: Scaling language modeling with pathways. *The Journal of Machine Learning Research*, 2023, 24(1): 240. DOI: [10.5555/3648699.3648939](https://doi.org/10.5555/3648699.3648939).
- [50] Lee K, Ippolito D, Nystrom A, Zhang C, Eck D, Callison-Burch C, Carlini N. Deduplicating training data makes language models better. In *Proc. the 60th Annual Meeting of the Association for Computational Linguistics*, May 2022, pp.8424–8445. DOI: [10.18653/v1/2022.acl-long.577](https://doi.org/10.18653/v1/2022.acl-long.577).
- [51] Kandpal N, Wallace E, Raffel C. Deduplicating training data mitigates privacy risks in language models. In *Proc. the 39th International Conference on Machine Learning*, Jul. 2022, pp.10697–10707.
- [52] Touvron H, Lavril T, Izacard G, Martinet X, Lachaux M A, Lacroix T, Roziere B, Goyal N, Hambro E, Azhar F, Rodriguez A, Joulin A, Grave E, Lample G. LLaMA: Open and efficient foundation language models. arXiv: 2302.13971, 2023. <https://arxiv.org/abs/2302.13971>, Jan. 2026.
- [53] Touvron H, Martin L, Stone K et al. LLAMA 2: Open foundation and fine-tuned chat models. arXiv: 2307.09288, 2023. <https://arxiv.org/abs/2307.09288>, Jan. 2026.
- [54] Kocetkov D, Li R, Ben Allal L, Li J, Mou C, Munoz Ferrandis C, Jernite Y, Mitchell M, Hughes S, Wolf T, Bahdanau D, von Werra L, de Vries H. The stack: 3 TB of permissively licensed source code. *Trans. Machine Learning Research*, 2023. <https://openreview.net/pdf?id=pxpbTdUEpD>, Jan. 2026.
- [55] Together. Redpajama-data-v2: An open dataset with 30 trillion tokens for training large language models. Together AI Blog, October 2023. <https://www.together.ai/blog/redpajama-data-v2>, Jan. 2026.
- [56] Manber U, Myers G. Suffix arrays: A new method for on-line string searches. *SIAM Journal on Computing*, 1993, 22(5): 935–948. DOI: [10.1137/0222058](https://doi.org/10.1137/0222058).
- [57] Landi I, Allewaert E, Valentine A A, Lepow L A, Charney A W. Clinical text deduplication practices for efficient pretraining and improved clinical tasks. arXiv: 2312.09469, 2023. <https://arxiv.org/abs/2312.09469>, Jan. 2026.
- [58] Broder A Z. On the resemblance and containment of documents. In *Proc. the 1997 Compression and Complexity of SEQUENCES*, Jun. 1997, pp.21–29. DOI: [10.1109/SEQUEN.1997.666900](https://doi.org/10.1109/SEQUEN.1997.666900).
- [59] Broder A Z, Glassman S C, Manasse M S, Zweig G. Syntactic clustering of the Web. *Computer Networks and ISDN Systems*, 1997, 29(8/9/10/11/12/13): 1157–1166. DOI: [10.1016/S0169-7552\(97\)00031-7](https://doi.org/10.1016/S0169-7552(97)00031-7).
- [60] 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. arXiv: 2101.00027, 2020. <https://arxiv.org/abs/2101.00027>, Jan. 2026.
- [61] Rae J W, Borgeaud S, Cai T et al. Scaling language models: Methods, analysis & insights from training gopher. arXiv: 2112.11446, 2022. <https://arxiv.org/abs/2112.11446>, Jan. 2026.
- [62] Shen Z, Tao T, Ma L, Neiswanger W, Liu Z, Wang H, Tan B, Hestness J, Vassilieva N, Soboleva D, Xing E. SlimPajama-DC: Understanding data combinations for LLM training. arXiv: 2309.10818, 2023. <https://arxiv.org/abs/2309.10818>, Jan. 2026.
- [63] Manku G S, Jain A, Das Sarma A. Detecting near-duplicates for web crawling. In *Proc. the 16th International Conference on World Wide Web*, May 2007, pp.141–150. DOI: [10.1145/1242572.1242592](https://doi.org/10.1145/1242572.1242592).
- [64] Chen Y, Cai W, Wu L, Li X, Xin Z, Fu C. TigerBot: An open multilingual multitask LLM. arXiv: 2312.08688, 2023. <https://arxiv.org/abs/2312.08688>, Jan. 2026.
- [65] He N, Xiong W, Liu H, Liao Y, Ding L, Zhang K, Tang G, Han X, Yang W. SoftDedup: An efficient data reweighting method for speeding up language model pre-training. In *Proc. the 62nd Annual Meeting of the Association for Computational Linguistics*, Aug. 2024, pp.4011–4022. DOI: [10.18653/v1/2024.acl-long.220](https://doi.org/10.18653/v1/2024.acl-long.220).
- [66] Khan A, Underwood R, Siebenshuh C, Babuji Y, Ajith A, Hippe K, Gokdemir O, Brace A, Chard K, Foster I. LSHBloom: Memory-efficient, extreme-scale document deduplication. arXiv: 2411.04257, 2024. <https://arxiv.org/abs/2411.04257>, Jan. 2026.
- [67] Shilov I, Meeus M, de Montjoye Y A. The mosaic memory of large language models. *Nature Communications*, 2026. DOI: [10.1038/s41467-026-68603-0](https://doi.org/10.1038/s41467-026-68603-0).
- [68] Abbas A, Tirumala K, Simig D, Ganguli S, Morcos A S. SemDeDup: Data-efficient learning at web-scale through semantic deduplication. arXiv: 2303.09540, 2023. <https://arxiv.org/abs/2303.09540>, Jan. 2026.
- [69] Tirumala K, Simig D, Aghajanyan A, Morcos A S. D4: Improving LLM pretraining via document deduplication and diversification. In *Proc. the 37th International Conference on Neural Information Processing Systems*, Dec. 2023, Article No. 2348. DOI: [10.5555/3666122.3668470](https://doi.org/10.5555/3666122.3668470).
- [70] Kaddour J. The MiniPile challenge for data-efficient language models. arXiv: 2304.08442, 2023. <https://arxiv.org/abs/2304.08442>, Jan. 2026.
- [71] Chen D, Huang Y, Ma Z, Chen H, Pan X, Ge C, Gao D, Xie Y, Liu Z, Gao J, Li Y, Ding B, Zhou J. Data-juicer:

- A one-stop data processing system for large language models. In *Proc. the 2024 International Conference on Management of Data*, Jun. 2024, pp.120–134. DOI: [10.1145/3626246.3653385](https://doi.org/10.1145/3626246.3653385).
- [72] Silcock E, D’Amico-Wong L, Yang J, Dell M. Noise-robust de-duplication at scale. In *Proc. the 11th International Conference on Learning Representations*, May 2023.
- [73] Reimers N, Gurevych I. Sentence-BERT: Sentence embeddings using siamese BERT-networks. In *Proc. the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Nov. 2019, pp.3982–3992. DOI: [10.18653/v1/D19-1410](https://doi.org/10.18653/v1/D19-1410).
- [74] Li X, Li J. Generative deduplication for socia media data selection. In *Proc. the 2024 Findings of the Association for Computational Linguistics*, Nov. 2024, pp.5765–5776. DOI: [10.18653/v1/2024.findings-emnlp.330](https://doi.org/10.18653/v1/2024.findings-emnlp.330).
- [75] Barbaresi A. Trafilatura: A web scraping library and command-line tool for text discovery and extraction. In *Proc. the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, Aug. 2021, pp.122–131. DOI: [10.18653/v1/2021.acl-demo.15](https://doi.org/10.18653/v1/2021.acl-demo.15).
- [76] Kohlschutter C, Fankhauser P, Nejdl W. Boilerplate detection using shallow text features. In *Proc. the 3rd ACM International Conference on Web Search and Data Mining*, Feb. 2010, pp.441–450. DOI: [10.1145/1718487.1718542](https://doi.org/10.1145/1718487.1718542).
- [77] Xie Y, Aggarwal K, Ahmad A. Efficient continual pre-training for building domain specific large language models. In *Proc. the 2024 Findings of the Association for Computational Linguistics*, Aug. 2024, pp.10184–10201. DOI: [10.18653/v1/2024.findings-acl.606](https://doi.org/10.18653/v1/2024.findings-acl.606).
- [78] Hirano M, Imajo K. Construction of domain-specified Japanese large language model for finance through continual pre-training. In *Proc. the 16th IIAI International Congress on Advanced Applied Informatics (IIAI-AAI)*, Jul. 2024, pp.273–279. DOI: [10.1109/IIAI-AAI63651.2024.00059](https://doi.org/10.1109/IIAI-AAI63651.2024.00059).
- [79] Almeida T S, Nogueira R, Pedrini H. Building high-quality datasets for portuguese LLMs: From common crawl snapshots to industrial-grade corpora. *Journal of the Brazilian Computer Society*, 2025, 31(1): 1246–1262. DOI: [10.5753/jbcs.2025.5788](https://doi.org/10.5753/jbcs.2025.5788).
- [80] Imani A, Lin P, Kargaran A H, Severini S, Sabet M J, Kassner N, Ma C, Schmid H, Martins A, Yvon F, Schütze H. Glot500: Scaling multilingual corpora and language models to 500 languages. In *Proc. the 61st Annual Meeting of the Association for Computational Linguistics*, Jul. 2023, pp.1082–1117. DOI: [10.18653/v1/2023.acl-long.61](https://doi.org/10.18653/v1/2023.acl-long.61).
- [81] Ji S, Li Z, Paul I, Paavola J, Lin P, Chen P, O’Brien D, Luo H, Schütze H, Tiedemann J, Haddow B. EMMA-500: Enhancing massively multilingual adaptation of large language models. arXiv: 2409.17892, 2024. <https://arxiv.org/abs/2409.17892>, Jan. 2026.
- [82] Dou L, Liu Q, Zeng G, Guo J, Zhou J, Mao X, Jin Z, Lu W, Lin M. Sailor: Open language models for south-east Asia. In *Proc. the 2024 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Nov. 2024, pp.424–435. DOI: [10.18653/v1/2024.emnlp-demo.45](https://doi.org/10.18653/v1/2024.emnlp-demo.45).
- [83] Nakamura T, Mishra M, Tedeschi S *et al.* Aurora-M: Open source continual pre-training for multilingual language and code. In *Proc. the 31st International Conference on Computational Linguistics: Industry Track*, Jan. 2025, pp.656–678.
- [84] Nag A, Chakrabarti S, Mukherjee A, Ganguly N. Efficient continual pre-training of LLMs for low-resource languages. In *Proc. the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies*, Apr. 2025, pp.304–317. DOI: [10.18653/v1/2025.naacl-industry.25](https://doi.org/10.18653/v1/2025.naacl-industry.25).
- [85] Jang J, Ye S, Lee C, Yang S, Shin J, Han J, Kim G, Seo M. TemporalWiki: A lifelong benchmark for training and evaluating ever-evolving language models. In *Proc. the 2022 Conference on Empirical Methods in Natural Language Processing*, Dec. 2022, pp.6237–6250. DOI: [10.18653/v1/2022.emnlp-main.418](https://doi.org/10.18653/v1/2022.emnlp-main.418).
- [86] Yu S, Liu Z, Xiong C. Craw4LLM: Efficient web crawling for LLM pretraining. In *Proc. the 2025 Findings of the Association for Computational Linguistics*, Jul. 2025, pp.13843–13851. DOI: [10.18653/v1/2025.findings-acl.712](https://doi.org/10.18653/v1/2025.findings-acl.712).
- [87] Xu Z, Liu Z, Yan Y, Liu Z, Yu G, Xiong C. Cleaner pre-training corpus curation with neural web scraping. In *Proc. the 62nd Annual Meeting of the Association for Computational Linguistics*, Aug. 2024, pp.802–812. DOI: [10.18653/v1/2024.acl-short.72](https://doi.org/10.18653/v1/2024.acl-short.72).
- [88] Guo Y, Fu J, Zhang H, Zhao D. Efficient domain continual pretraining by mitigating the stability gap. In *Proc. the 63rd Annual Meeting of the Association for Computational Linguistics*, Jul. 2025, pp.32850–32870. DOI: [10.18653/v1/2025.acl-long.1578](https://doi.org/10.18653/v1/2025.acl-long.1578).
- [89] Wang Y, Fu Z, Cai J, Tang P, Lyu H, Fang Y, Zheng Z, Zhou J, Zeng G, Xiao C, Han X, Liu Z. Ultra-FineWeb: Efficient data filtering and verification for high-quality LLM training data. arXiv: 2505.05427, 2025. <https://arxiv.org/abs/2505.05427>, Jan. 2026.
- [90] Shi W, Min S, Lomeli M, Zhou C, Li M, Lin X V, Smith N A, Zettlemoyer L, Yih W T, Lewis M. In-context pre-training: Language modeling beyond document boundaries. In *Proc. the 2024 International Conference on Learning Representations*, May 2024.
- [91] Parmar J, Satheesh S, Patwary M, Shoneybi M, Catanzaro B. Reuse, don’t retrain: A recipe for continued pre-training of language models. arXiv: 2407.07263, 2024. <https://arxiv.org/abs/2407.07263>, Jan. 2026.
- [92] Sam D, Chakrabarti A, Rostamizadeh A, Ramalingam S,

- Citovsky G, Kumar S. Analyzing similarity metrics for data selection for language model pretraining. arXiv: 2502.02494, 2025. <https://arxiv.org/abs/2502.02494>, Feb. 2026.
- [93] Mizrahi D, Larsen A B L, Allardice J, Petryk S, Gorokhov Y, Li J, Fang A, Gardner J, Gunter T, Dehghan A. Language models improve when pretraining data matches target tasks. arXiv: 2507.12466, 2025. <https://arxiv.org/abs/2507.12466>, Jan. 2026.
- [94] Yauney G, Reif E, Mimno D. Data similarity is not enough to explain language model performance. In *Proc. the 2023 Conference on Empirical Methods in Natural Language Processing*, Dec. 2023, pp.11295–11304. DOI: [10.18653/v1/2023.emnlp-main.695](https://doi.org/10.18653/v1/2023.emnlp-main.695).
- [95] Yildiz C, Ravichandran N K, Sharma N, Bethge M, Ermis B. Investigating continual pretraining in large language models: Insights and implications. arXiv: 2402.17400, 2024. <https://arxiv.org/abs/2402.17400>, Jan. 2026.
- [96] Wang X, Tissue H, Wang L, Li L, Zeng D D. Learning dynamics in continual pre-training for large language models. In *Proc. the 42nd International Conference on Machine Learning*, Jul. 2025.
- [97] Chen J, Chen Z, Wang J et al. Towards effective and efficient continual pre-training of large language models. In *Proc. the 63rd Annual Meeting of the Association for Computational Linguistics*, Jul. 2025, pp.5779–5795. DOI: [10.18653/v1/2025.acl-long.289](https://doi.org/10.18653/v1/2025.acl-long.289).
- [98] Pan R, Nguyen T D, Arora H, Accomazzi A, Ghosal T, Ting Y S. AstroMLab 2: AstroLLaMA-2-70B model and benchmarking specialised LLMs for astronomy. In *Proc. the 2024 SC Workshops of the International Conference on High Performance Computing, Network, Storage, and Analysis*, Nov. 2024, pp.87–96. DOI: [10.1109/SCW63240.2024.00019](https://doi.org/10.1109/SCW63240.2024.00019).
- [99] Maini P, Seto S, Bai H, Grangier D, Zhang Y, Jaitly N. Rephrasing the web: A recipe for compute and data-efficient language modeling. In *Proc. the 62nd Annual Meeting of the Association for Computational Linguistics*, Aug. 2024, pp.14044–14072. DOI: [10.18653/v1/2024.acl-long.757](https://doi.org/10.18653/v1/2024.acl-long.757).
- [100] Jiang J, Li J, Zhao X, Song Y, Zhang T, Wen J R. Mix-CPT: A domain adaptation framework via decoupling knowledge learning and format alignment. In *Proc. the 13th International Conference on Learning Representations*, Apr. 2025.
- [101] Hu Y, Song H, Deng J, Wang J, Chen J, Zhou K, Zhu Y, Jiang J, Dong Z, Zhao W X, Wen J R. YuLan-Mini: An open data-efficient language model. arXiv: 2412.17743, 2024. <https://arxiv.org/abs/2412.17743>, Jan. 2026.
- [102] Abdin M, Aneja J, Behl H et al. Phi-4 technical report. arXiv: 2412.08905, 2024. <https://arxiv.org/abs/2412.08905>, Jan. 2026.
- [103] Ishibashi Y, Yano T, Oyamada M. Mining hidden thoughts from texts: Evaluating continual pretraining with synthetic data for LLM reasoning. arXiv: 2505.10182, 2025. <https://arxiv.org/abs/2505.10182>, Feb. 2026.
- [104] Yang Z, Band N, Li S, Candes E J, Hashimoto T. Synthetic continued pretraining. In *Proc. the 13th International Conference on Learning Representations*, Apr. 2025.
- [105] Akter S N, Prabhumoye S, Kamalu J, Satheesh S, Nyberg E, Patwary M, Shoeybi M, Catanzaro B. MIND: Math informed synthetic dialogues for pretraining LLMs. In *Proc. the 13th International Conference on Learning Representations*, Apr. 2025.
- [106] Dong H, Zhang P, Lu M, Shen Y, Ke G. MachineLearningLM: Continued pretraining language models on millions of synthetic tabular prediction tasks scales in-context ML. arXiv: 2509.06806, 2025. <https://arxiv.org/abs/2509.06806v1>, Jan. 2026.
- [107] Liang X, Hu X, Zuo S, Gong Y, Lou Q, Liu Y, Huang S L, Jiao J. Task oriented in-domain data augmentation. In *Proc. the 2024 Conference on Empirical Methods in Natural Language Processing*, Nov. 2024, pp.20889–20907. DOI: [10.18653/v1/2024.emnlp-main.1162](https://doi.org/10.18653/v1/2024.emnlp-main.1162).
- [108] Mishra S, Khashabi D, Baral C, Hajishirzi H. Cross-task generalization via natural language crowdsourcing instructions. In *Proc. the 60th Annual Meeting of the Association for Computational Linguistics*, May 2022, pp.3470–3487. DOI: [10.18653/v1/2022.acl-long.244](https://doi.org/10.18653/v1/2022.acl-long.244).
- [109] Wei J, Bosma M, Zhao V Y, Guu K, Yu A W, Lester B, Du N, Dai A M, Le Q V. Finetuned language models are zero-shot learners. In *Proc. the 10th International Conference on Learning Representations*, Apr. 2022.
- [110] Bach S H, Sanh V, Yong Z X et al. PromptSource: An integrated development environment and repository for natural language prompts. In *Proc. the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, May 2022, pp.93–104. DOI: [10.18653/v1/2022.acl-demo.9](https://doi.org/10.18653/v1/2022.acl-demo.9).
- [111] Zhou C, Liu P, Xu P, Iyer S, Sun J, Mao Y, Ma X, Efrat A, Yu P, Yu L, Zhang S, Ghosh G, Lewis M, Zettlemoyer L, Levy O. LIMA: Less is more for alignment. In *Proc. the 37th International Conference on Neural Information Processing Systems*, Dec. 2023, Article No. 2400. DOI: [10.5555/3666122.3668522](https://doi.org/10.5555/3666122.3668522).
- [112] Ouyang L, Wu J, Jiang X et al. Training language models to follow instructions with human feedback. In *Proc. the 36th International Conference on Neural Information Processing Systems*, Nov. 28–Dec. 9, 2022, Article No. 2011. DOI: [10.5555/3600270.3602281](https://doi.org/10.5555/3600270.3602281).
- [113] Köpf A, Kilcher Y, von Rütte D et al. Openassistant conversations-democratizing large language model alignment. In *Proc. the 37th International Conference on Neural Information Processing Systems*, Dec. 2023, Article No. 2064. DOI: [10.5555/3666122.3668186](https://doi.org/10.5555/3666122.3668186).
- [114] Singh S, Vargus F, D’souza D et al. Aya dataset: An open-access collection for multilingual instruction tuning. In *Proc. the 62nd Annual Meeting of the Association for*

- Computational Linguistics*, Aug. 2024, pp.11521–11567. DOI: [10.18653/v1/2024.acl-long.620](https://doi.org/10.18653/v1/2024.acl-long.620).
- [115] Zhu H, Ding Y, Tao Y, Ruan Z, Li Y, Zhang W, Chen Y, Chen G. FANNO: Augmenting high-quality instruction data with open-sourced LLMs only. In *Findings of the Association for Computational Linguistics: ACL 2025*, Jul. 2025, pp.17633–17653. DOI: [10.18653/v1/2025.findings-acl.906](https://doi.org/10.18653/v1/2025.findings-acl.906).
- [116] Zhu A, Asawa P, Quincy Davis J, Chen L, Hanin B, Stolica I, Gonzalez J E, Zaharia M. BARE: Combining base and instruction-tuned language models for better synthetic data generation. arXiv: 2502.01697, 2025. <https://arxiv.org/abs/2502.01697v1>, Jan. 2026.
- [117] Tang X, Klein J, Bissyande T F. Boosting open-source LLMs or program repair via reasoning transfer and LLM-guided reinforcement learning. arXiv: 2506.03921, 2025. <https://arxiv.org/abs/2506.03921>, Jan. 2026.
- [118] He Y, Yin D, Peng N. Guiding through complexity: What makes good supervision for hard reasoning tasks. In *Proc. the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1)*, Apr. 2025, pp.11006–11046. DOI: [10.18653/v1/2025.naacl-long.550](https://doi.org/10.18653/v1/2025.naacl-long.550).
- [119] Dong J, Jiang L, Jin W, Cheng L. Threshold filtering packing for supervised finetuning. In *Proc. the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1)*, Apr. 2025, pp.4422–4435. DOI: [10.18653/v1/2025.naacl-long.226](https://doi.org/10.18653/v1/2025.naacl-long.226).
- [120] Wang S, Jin X, Wang Z, Wang J, Zhang J, Li K, Wen Z, Li Z, He C, Hu X, Zhang L. Data whisperer: Efficient data selection for task-specific LLM fine-tuning via few-shot in-context learning. In *Proc. the 63rd Annual Meeting of the Association for Computational Linguistics*, Jul. 2025, pp.23287–23305. DOI: [10.18653/v1/2025.acl-long.1135](https://doi.org/10.18653/v1/2025.acl-long.1135).
- [121] Zhang J, Zhang C X, Liu Y, Jin Y X, Yang X W, Zheng B, Liu Y, Guo L Z. D3: Diversity, difficulty, and dependability-aware data selection for sample-efficient LLM instruction tuning. In *Proc. the 34th International Joint Conference on Artificial Intelligence*, Aug. 2025, Article No. 928. DOI: [10.24963/ijcai.2025/928](https://doi.org/10.24963/ijcai.2025/928).
- [122] Nikdan M, Cohen-Addad V, Alistarh D, Mirrokni V. Efficient data selection at scale via influence distillation. arXiv: 2505.19051, 2025. <https://arxiv.org/abs/2505.19051>, Jan. 2026.
- [123] Sener O, Savarese S. Active learning for convolutional neural networks: A core-set approach. In *Proc. the 6th International Conference on Learning Representations*, Apr. 30–May 3, 2018, pp.2232–2244.
- [124] Kothawade S, Beck N, Killamsetty K, Iyer R. SIMILAR: Submodular information measures based active learning in realistic scenarios. In *Proc. the 35th International Conference on Neural Information Processing Systems*, Dec. 2021, Article No. 1428. DOI: [10.5555/3540261.3541689](https://doi.org/10.5555/3540261.3541689).
- [125] Wei K, Iyer R, Bilmes J. Submodularity in data subset selection and active learning. In *Proc. the 32nd International Conference on Machine Learning*, Jul. 2015, pp.1954–1963. DOI: [10.5555/3045118.3045326](https://doi.org/10.5555/3045118.3045326).
- [126] Bachem O, Lucic M, Krause A. Practical coresets constructions for machine learning. arXiv: 1703.06476, 2017. <https://arxiv.org/abs/1703.06476>, Jan. 2026.
- [127] Feldman D, Langberg M. A unified framework for approximating and clustering data. In *Proc. the 43rd Annual ACM Symposium on Theory of Computing*, Jun. 2011, pp.569–578. DOI: [10.1145/1993636.1993712](https://doi.org/10.1145/1993636.1993712).
- [128] Fu J, Ng S K, Jiang Z, Liu P. GPTScore: Evaluate as you desire. In *Proc. the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Jun. 2024, pp.6556–6576. DOI: [10.18653/v1/2024.naacl-long.365](https://doi.org/10.18653/v1/2024.naacl-long.365).
- [129] Du Q, Zong C, Zhang J. MoDS: Model-oriented data selection for instruction tuning. arXiv: 2311.15653, 2023. <https://arxiv.org/abs/2311.15653>, Jan. 2026.
- [130] Parkar R S, Kim J, Park J I, Kang D. SelectLLM: Can LLMs select important instructions to annotate? arXiv: 2401.16553, 2024. <https://arxiv.org/abs/2401.16553>, Jan. 2026.
- [131] Madaan A, Tandon N, Gupta P et al. SELF-REFINE: Iterative refinement with self-feedback. In *Proc. the 37th International Conference on Neural Information Processing Systems*, Dec. 2023, pp.46534–46594. DOI: [10.5555/3666122.3668141](https://doi.org/10.5555/3666122.3668141).
- [132] Chu K C, Chen Y P, Nakayama H. A better LLM evaluator for text generation: The impact of prompt output sequencing and optimization. arXiv: 2406.09972, 2024. <https://arxiv.org/abs/2406.09972>, Jan. 2026.
- [133] Lee J, Hockenmaier J. Evaluating step-by-step reasoning traces: A survey. In *Proc. the 2023 Findings of the Association for Computational Linguistics*, Nov. 2025, pp.1789–1814. DOI: [10.18653/v1/2025.findings-emnlp.94](https://doi.org/10.18653/v1/2025.findings-emnlp.94).
- [134] Paul M, Ganguli S, Dziugaite G K. Deep learning on a data diet: Finding important examples early in training. In *Proc. the 2023 Findings of the Association for Computational Linguistics*, Dec. 2021, pp.20596–20607. DOI: [10.5555/3540261.3541836](https://doi.org/10.5555/3540261.3541836).
- [135] Attenu J M, Corbeil J P. NLU on data diets: Dynamic data subset selection for NLP classification tasks. In *Proc. the 4th Workshop on Simple and Efficient Natural Language Processing (SustaiNLP)*, Jul. 2023, pp.129–146. DOI: [10.18653/v1/2023.sustainlp-1.9](https://doi.org/10.18653/v1/2023.sustainlp-1.9).
- [136] Xia M, Malladi S, Gururangan S, Arora S, Chen D. Less: Selecting influential data for targeted instruction tuning. In *Proc. the 41st International Conference on Machine Learning*, Jul. 2024, pp.11727–11737.
- [137] Choe S K, Ahn H, Bae J, Zhao K, Kang M, Chung Y, Pratapa A, Neiswanger W, Strubell E, Mitamura T, Schneider J, Hovy E, Grosse R, Xing E. What is your data worth to GPT? LLM-scale data valuation with influence functions. arXiv: 2405.13954, 2024. <https://arxiv.org/abs/2405.13954>.

- [org/abs/2405.13954](https://arxiv.org/abs/2405.13954), Jan. 2026.
- [138] Ananta M, Adilazuarda M F, Zuhri Z M K, Purwarianti A, Fikri Aji A. QLESS: A quantized approach for data valuation and selection in large language model fine-tuning. arXiv: 2502.01703, 2025. <https://arxiv.org/abs/2502.01703>, Jan. 2026.
- [139] Li M, Zhang Y, Li Z, Chen J, Chen L, Cheng N, Wang J, Zhou T, Xiao J. From quantity to quality: Boosting LLM performance with self-guided data selection for instruction tuning. In *Proc. the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Jun. 2024, pp.7602–7635. DOI: [10.18653/v1/2024.naacl-long.421](https://doi.org/10.18653/v1/2024.naacl-long.421).
- [140] Li Y, Hui B, Xia X, Yang J, Yang M, Zhang L, Si S, Chen L H, Liu J, Liu T, Huang F, Li Y. One-shot learning as instruction data prospector for large language models. In *Proc. the 62nd Annual Meeting of the Association for Computational Linguistics*, Aug. 2024, pp.4586–4601. DOI: [10.18653/v1/2024.acl-long.252](https://doi.org/10.18653/v1/2024.acl-long.252).
- [141] Kung P N, Yin F, Wu D, Chang K W, Peng N. Active instruction tuning: Improving cross-task generalization by training on prompt sensitive tasks. In *Proc. the 2023 Conference on Empirical Methods in Natural Language Processing*, Dec. 2023, pp.1813–1829. DOI: [10.18653/v1/2023.emnlp-main.112](https://doi.org/10.18653/v1/2023.emnlp-main.112).
- [142] Liu L, Liu X, Wong D F, Li D, Wang Z, Hu B, Zhang M. SelectIT: Selective instruction tuning for large language models via uncertainty-aware self-reflection. arXiv: 2402.16705, 2024. <https://arxiv.org/abs/2402.16705v1>, Jan. 2026.
- [143] See A, Liu P J, Manning C D. Get to the point: Summarization with pointer-generator networks. In *Proc. the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Jul. 2017, pp.1073–1083. DOI: [10.18653/v1/P17-1099](https://doi.org/10.18653/v1/P17-1099).
- [144] Narayan S, Cohen S B, Lapata M. Don't give me the details, just the summary! Topic-aware convolutional neural networks for extreme summarization. In *Proc. the 2018 Conference on Empirical Methods in Natural Language Processing*, Oct. 2018, pp.1797–1807. DOI: [10.18653/v1/D18-1206](https://doi.org/10.18653/v1/D18-1206).
- [145] Huang X Y, Vishnubhotla K, Rudzicz F. The GPT-WritingPrompts dataset: A comparative analysis of character portrayal in short stories. arXiv: 2406.16767, 2024. <https://arxiv.org/abs/2406.16767>, Jan. 2026.
- [146] Kumar N A, Pham C M, Iyyer M, Lan A. Whose story is it? Personalizing story generation by inferring author styles. In *Proc. the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, Dec. 2025, pp.1485–1540.
- [147] Rajpurkar P, Zhang J, Lopyrev K, Liang P. SQuAD: 100, 000+ questions for machine comprehension of text. In *Proc. the 2016 Conference on Empirical Methods in Natural Language Processing*, Nov. 2016, pp.2383–2392. DOI: [10.18653/v1/D16-1264](https://doi.org/10.18653/v1/D16-1264).
- [148] Kwiatkowski T, Palomaki J, Redfield O et al. Natural Questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 2019, 7:452–466. DOI: [10.1162/tac1\\_a\\_00276](https://doi.org/10.1162/tac1_a_00276).
- [149] Joshi M, Choi E, Weld D S, Zettlemoyer L. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proc. the 55th Annual Meeting of the Association for Computational Linguistics*, Jul. 2017, pp.1601–1611. DOI: [10.18653/v1/P17-1147](https://doi.org/10.18653/v1/P17-1147).
- [150] Wang X, Zhou W, Zu C, Xia H, Chen T, Zhang Y, Zheng R, Ye J, Zhang Q, Gui T, Kang J, Yang J, Li S, Du C. InstructUIE: Multi-task instruction tuning for unified information extraction. arXiv: 2304.08085, 2023. <https://arxiv.org/abs/2304.08085>, Jan. 2026.
- [151] Zhang Z, You W, Wu T, Wang X, Li J, Zhang M. A survey of generative information extraction. In *Proc. the 31st International Conference on Computational Linguistics*, Jan. 2025, pp.4840–4870.
- [152] Wei J, Wang X, Schuurmans D, Bosma M, Ichter B, Xia F, Chen E H, Le Q V, Zhou D. Chain-of-thought prompting elicits reasoning in large language models. In *Proc. the 36th International Conference on Neural Information Processing Systems*, Nov. 28-Dec. 9, 2022, Article No. 1800. DOI: [10.5555/3600270.3602070](https://doi.org/10.5555/3600270.3602070).
- [153] Cobbe K, Kosaraju V, Bavarian M, Chen M, Jun H, Kaiser L, Plappert M, Tworek J, Hilton J, Nakano R, Hesse C, Schulman J. Training verifiers to solve math word problems. arXiv: 2110.14168, 2021. <https://arxiv.org/abs/2110.14168>, Jan. 2026.
- [154] Hendrycks D, Burns C, Kadavath S, Arora A, Basart S, Tang E, Song D, Steinhardt J. Measuring mathematical problem solving with the MATH dataset. In *Proc. the 2021 Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks*, Dec. 2021.
- [155] Chen M, Tworek J, Jun H et al. Evaluating large language models trained on code. arXiv: 2107.03374, 2021. <https://arxiv.org/abs/2107.03374>, Jan. 2026.
- [156] Austin J, Odena A, Nye M, Bosma M, Michalewski H, Dohan D, Jiang E, Cai C, Terry M, Le Q, Sutton C. Program synthesis with large language models. arXiv: 2108.07732, 2021. <https://arxiv.org/abs/2108.07732>, Jan. 2026.
- [157] Gao L, Madaan A, Zhou S, Alon U, Liu P, Yang Y, Callan J, Neubig G. PAL: Program-aided language models. In *Proc. the 40th International Conference on Machine Learning*, Jul. 2023, pp.10764–10799.
- [158] Chen W, Ma X, Wang X, Cohen W W. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Trans. Machine Learning Research*, 2023. <https://openreview.net/pdf?id=YfZ4ZPt8zd>, Jan. 2026.
- [159] Geva M, Khashabi D, Segal E, Khot T, Roth D, Berant J. Did Aristotle use a laptop? A question answering benchmark with implicit reasoning strategies. *Transac-*

- tions of the Association for Computational Linguistics, 2021, 9:346–361. DOI: [10.1162/tac1\\_a\\_00370](https://doi.org/10.1162/tac1_a_00370).
- [160] Chen W, He W, Xi Z, Guo H, Hong B, Zhang J, Li N, Gui T, Li Y, Zhang Q, Huang X. Better process supervision with bi-directional rewarding signals. In *Proc. the 2025 Findings of the Association for Computational Linguistics*, Jul. 2025, pp.14471–14485. DOI: [10.18653/v1/2025.findings-acl.747](https://doi.org/10.18653/v1/2025.findings-acl.747).
- [161] Puerto H, Chubakov T, Zhu X, Madabushi H T, Gurevych I. Fine-tuning on diverse reasoning chains drives within-inference CoT refinement in LLMs. In *Proc. the 63rd Annual Meeting of the Association for Computational Linguistics*, Jul. 2025, pp.3789–3808. DOI: [10.18653/v1/2025.acl-long.191](https://doi.org/10.18653/v1/2025.acl-long.191).
- [162] Sanh V, Webson A, Raffel C *et al.* Multitask prompted training enables zero-shot task generalization. In *Proc. the 10th International Conference on Learning Representations*, Apr. 2022.
- [163] Wang A, Singh A, Michael J, Hill F, Levy O, Bowman S. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proc. the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, Nov. 2018, pp.353–355. DOI: [10.18653/v1/W18-5446](https://doi.org/10.18653/v1/W18-5446).
- [164] Wang A, Pruksachatkun Y, Nangia N, Singh A, Michael J, Hill F, Levy O, Bowman S R. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In *Proc. the 33rd International Conference on Neural Information Processing Systems*, Dec. 2019, Article No. 294. DOI: [10.5555/3454287.3454581](https://doi.org/10.5555/3454287.3454581).
- [165] He B, Ding N, Qian C, Deng J, Cui G, Yuan L, Gao H A, Chen H, Liu Z, Sun M. Zero-shot generalization during instruction tuning: Insights from similarity and granularity. arXiv: 2406.11721, 2024. <https://arxiv.org/abs/2406.11721v1>, Jan. 2026.
- [166] Zhao Y, Muraoka M, Yoshida I, Bhattacharjee B, Kanayama H. A simple-yet-efficient instruction augmentation method for zero-shot sentiment classification. In *Proc. the 31st International Conference on Computational Linguistics*, Jan. 2025, pp.1585–1599.
- [167] The Vicuna Team. Vicuna: An open-source chatbot impressing GPT-4 with 90%\* chatGPT quality, 2023. <https://lmsys.org/blog/2023-03-30-vicuna/>, February 2026.
- [168] Chung H W, Hou L, Longpre S *et al.* Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 2024, 25(1): Article No. 70.
- [169] Bai Y, Jones A, Ndousse K *et al.* Training a helpful and harmless assistant with reinforcement learning from human feedback. arXiv: 2204.05862, 2022. <https://arxiv.org/abs/2204.05862>, Jan. 2026.
- [170] Xu C, Guo D, Duan N, McAuley J. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. In *Proc. the 2023 Conference on Empirical Methods in Natural Language Processing*, Dec. 2023, pp.6268–6278. DOI: [10.18653/v1/2023.emnlp-main.385](https://doi.org/10.18653/v1/2023.emnlp-main.385).
- [171] Kumar B, Amar J, Yang E, Li N, Jia Y. Selective fine-tuning on LLM-labeled data may reduce reliance on human annotation: A case study using schedule-of-event table detection. In *Proc. the 9th Machine Learning for Healthcare Conference*, Aug. 2024.
- [172] Ge Y, Liu Y, Ye Z, Mao Y, Gao Y. Text-to-pipeline: Bridging natural language and data preparation pipelines. arXiv: 2505.15874, 2025. <https://arxiv.org/abs/2505.15874>, Jan. 2026.
- [173] Liang H, Ma X, Liu Z *et al.* DataFlow: An LLM-driven framework for unified data preparation and workflow automation in the era of data-centric AI. arXiv: 2512.16676, 2025. <https://arxiv.org/abs/2512.16676>, Jan. 2026.
- [174] Ghosh D, Chan S. LL-instruct: An instruction-tuned model for English language proficiency assessments. arXiv: 2410.09314, 2024. <https://arxiv.org/abs/2410.09314>, Jan. 2026.
- [175] Wang Y, Kordi Y, Mishra S, Liu A, Smith N A, Khashabi D, Hajishirzi H. Self-instruct: Aligning language models with self-generated instructions. In *Proc. the 61st Annual Meeting of the Association for Computational Linguistics*, Jul. 2023, pp.13484–13508. DOI: [10.18653/v1/2023.acl-long.754](https://doi.org/10.18653/v1/2023.acl-long.754).
- [176] Xu C, Sun Q, Zheng K, Geng X, Zhao P, Feng J, Tao C, Lin Q, Jiang D. WizardLM: Empowering large pre-trained language models to follow complex instructions. In *Proc. the 12th International Conference on Learning Representations*, May 2024.
- [177] Kaur S, Park S, Goyal A, Arora S. Instruct-SkillMix: A powerful pipeline for LLM instruction tuning. In *Proc. the 13th International Conference on Learning Representations*, Apr. 2025.
- [178] Li H, Dong Q, Tang Z *et al.* Synthetic data (almost) from scratch: Generalized instruction tuning for language models. arXiv: 2402.13064, 2024. <https://arxiv.org/abs/2402.13064>, Jan. 2026.
- [179] Xiao J, Chen Y, Ou Y, Yu H, Shu K, Xiao Y. Baichuan2-sum: Instruction finetune baichuan2-7B model for dialogue summarization. In *Proc. the 2024 International Joint Conference on Neural Networks (IJCNN)*, Jun. 30–Jul. 5, 2024. DOI: [10.1109/IJCNN60899.2024.10650513](https://doi.org/10.1109/IJCNN60899.2024.10650513).
- [180] Razumovskaina E, Glavaš G, Korhonen A, Vulić I. SQATIN: Supervised instruction tuning meets question answering for improved dialogue NLU. In *Proc. the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Jun. 2024, pp.8195–8211. DOI: [10.18653/v1/2024.naacl-long.453](https://doi.org/10.18653/v1/2024.naacl-long.453).
- [181] Hou X, Li Q, Yang J, Li T, Chai L, Wu X, Ji H, Li Z, Nie J, Dun J, Song W. Raw text is all you need: Knowledge-intensive multi-turn instruction tuning for large language model. arXiv: 2407.03040, 2024. <https://arxiv.org/abs/2407.03040>, Jan. 2026.
- [182] Shao Z, Wang P, Zhu Q, Xu R, Song J, Bi X, Zhang H, Zhang M, Li Y K, Wu Y, Guo D. DeepSeekMath: Push-

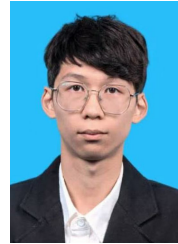
- ing the limits of mathematical reasoning in open language models. arXiv: 2402.03300, 2024. <https://arxiv.org/abs/2402.03300>, Jan. 2026.
- [183] Guo D, Yang D, Zhang H *et al.* DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. arXiv: 2501.12948, 2025. <https://arxiv.org/abs/2501.12948>, Jan. 2026.
- [184] Yu Q, Zhang Z, Zhu R *et al.* DAPO: An open-source LLM reinforcement learning system at scale. arXiv: 2503.14476, 2025. <https://arxiv.org/abs/2503.14476>, Jan. 2026.
- [185] Bae S, Hong J, Lee M Y, Kim H, Nam J, Kwak D. Online difficulty filtering for reasoning oriented reinforcement learning. arXiv: 2504.03380, 2025. <https://arxiv.org/abs/2504.03380>, Jan. 2026.
- [186] Dong H, Xiong W, Pang B, Wang H, Zhao H, Zhou Y, Jiang N, Sahoo D, Xiong C, Zhang T. RLHF workflow: From reward modeling to online RLHF. arXiv:2405.07863, 2024. <https://arxiv.org/html/2405.07863v3>, Jan. 2026.
- [187] Jin D, Mehri S, Hazarika D, Padmakumar A, Lee S, Liu Y, Namazifar M. Data-efficient alignment of large language models with human feedback through natural language. arXiv: 2311.14543, 2023. <https://arxiv.org/abs/2311.14543>, Feb. 2023.
- [188] Zhang G, Duan J. VickreyFeedback: Cost-efficient data construction for reinforcement learning from human feedback. In *Proc. the 25th International Conference on Principles and Practice of Multi-Agent Systems*, Nov. 2024, pp.351–366. DOI: [10.1007/978-3-031-77367-9\\_27](https://doi.org/10.1007/978-3-031-77367-9_27).
- [189] Morimura T, Sakamoto M, Jinnai Y, Abe K, Ariu K. Filtered direct preference optimization. In *Proc. the 2024 Conference on Empirical Methods in Natural Language Processing*, Nov. 2024, pp.22729–22770. DOI: [10.18653/v1/2024.emnlp-main.1266](https://doi.org/10.18653/v1/2024.emnlp-main.1266).
- [190] Xu Y, Chakraborty T, Kiciman E, Aryal B, Rodrigues E, Sharma S, Estevao R, de Luis Balaguer M A, Wolk J, Padilha R, Nunes L, Balakrishnan S, Lu S, Chandra R. RLTHF: Targeted human feedback for LLM alignment. arXiv: 2502.13417, 2025. <https://arxiv.org/abs/2502.13417>, Jan. 2026.
- [191] Wang B, Zheng R, Chen L, Xi Z, Shen W, Zhou Y, Yan D, Gui T, Zhang Q, Huang X. Reward modeling requires automatic adjustment based on data quality. In *Proc. the 2024 Findings of the Association for Computational Linguistics*, Nov. 2024, pp.4041–4064. DOI: [10.18653/v1/2024.findings-emnlp.234](https://doi.org/10.18653/v1/2024.findings-emnlp.234).
- [192] Shen L, Chen S, Song L, Jin L, Peng B, Mi H, Khashabi D, Yu D. The trickle-down impact of reward (in-)consistency on RLHF. arXiv: 2309.16155, 2023. <https://arxiv.org/abs/2309.16155>, Jan. 2026.
- [193] Lai Y, Wang S, Liu S, Huang X, Wei Z. ALaRM: Align language models via hierarchical rewards modeling. In *Proc. the 2024 Findings of the Association for Computational Linguistics*, Aug. 2024, pp.7817–7831. DOI: [10.18653/v1/2024.findings-acl.465](https://doi.org/10.18653/v1/2024.findings-acl.465).
- [194] Lambert N, Pyatkin V, Morrison J, Miranda L J, Lin B Y, Chandu K, Dziri N, Kumar S, Zick T, Choi Y, Smith N A, Hajishirzi H. RewardBench: Evaluating reward models for language modeling. In *Proc. the 2025 Findings of the Association for Computational Linguistics*, Apr. 2025, pp.1755–1797. DOI: [10.18653/v1/2025.findings-naacl.96](https://doi.org/10.18653/v1/2025.findings-naacl.96).
- [195] Liu Y, Yi X, Chen X, Yao J, Yi J, Zan D, Liu Z, Xie X, Ho T Y. Elephant in the room: Unveiling the impact of reward model quality in alignment. arXiv: 2409.19024, 2024. <https://arxiv.org/abs/2409.19024>, Jan. 2026.
- [196] Zhou C, Liu S, Wang Z, Wang D, Tu R C, Du B, Tao D. Intra-trajectory consistency for reward modeling. arXiv: 2506.09096, 2025. <https://arxiv.org/abs/2506.09096>, Jan. 2026.
- [197] Bai Y, Kadavath S, Kundu S *et al.* Constitutional AI: Harmlessness from AI feedback. arXiv: 2212.08073, 2022. <https://arxiv.org/abs/2212.08073>, Jan. 2026.
- [198] Lee H, Phatale S, Mansoor H, Mesnard T, Ferret J, Lu K, Bishop C, Hall E, Carbune V, Rastogi A, Prakash S. RLAIIF vs. RLHF: Scaling reinforcement learning from human feedback with AI feedback. In *Proc. the 41st International Conference on Machine Learning*, Jul. 2024.
- [199] Yuan W, Pang R Y, Cho K, Li X, Sukhbaatar S, Xu J, Weston J. Self-rewarding language models. In *Proc. the 41st International Conference on Machine Learning*, Jul. 2024.
- [200] Zheng L, Chiang W L, Sheng Y *et al.* Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Proc. the 37th International Conference on Neural Information Processing Systems*, Dec. 2023, pp.46595–46623. DOI: [10.5555/3666122.3668142](https://doi.org/10.5555/3666122.3668142).
- [201] Gu J, Jiang X, Shi Z, Tan H, Zhai X, Xu C, Li W, Shen Y, Ma S, Liu H, Wang S, Zhang K, Lin Z, Zhang B, Ni L, Gao W, Wang Y, Guo J. A survey on LLM-as-a-judge. *The Innovation*, 2026: 101253. DOI: [10.1016/j.xinn.2025.101253](https://doi.org/10.1016/j.xinn.2025.101253).
- [202] Wang Z, Zhou F, Li X, Liu P. OctoThinker: Mid-training incentivizes reinforcement learning scaling. arXiv: 2506.20512, 2025. <https://arxiv.org/abs/2506.20512>, Jan. 2026.
- [203] Tokpanov Y, Glorioso P, Anthony Q, Millidge B. Zyd-2: A 5 trillion token high-quality dataset. arXiv: 2411.06068, 2024. <https://arxiv.org/abs/2411.06068>, Jan. 2026.
- [204] Arefyev N, Aulamo M, Bañón M *et al.* HPLT’s second data release. In *Proc. the 2025 Machine Translation Summit XX*, Jun. 2025, pp.101–102.
- [205] Lozhkov A, Li R, Allal L B *et al.* Starcoder 2 and the Stack v2: The next generation. arXiv: 2402.19173, 2024. <https://arxiv.org/abs/2402.19173>, Jan. 2026.
- [206] Li J, Du L, Zhao H, Zhang B W, Wang L, Gao B, Liu G, Lin Y. Infinity instruct: Scaling instruction selection and synthesis to enhance language models. arXiv: 2506.11116, 2025. <https://arxiv.org/abs/2506.11116>, Jan. 2026.
- [207] Wang G, Cheng S, Zhan X, Li X, Song S, Liu Y. Open-

Chat: Advancing open-source language models with mixed-quality data. In *Proc. the 12th International Conference on Learning Representations*, May 2024.

- [208] Li J, Beeching E, Tunstall L *et al.* NuminaMath: The largest public dataset in AI4Maths with 860k pairs of competition math problems and solutions. *Hugging Face Repository*, 2024. [http://faculty.bicmr.pku.edu.cn/~dongbin/Publications/numina\\_dataset.pdf](http://faculty.bicmr.pku.edu.cn/~dongbin/Publications/numina_dataset.pdf), Jan. 2026.
- [209] Hosseini A, Sordoni A, Toyama D, Courville A, Agarwal R. Not all LLM reasoners are created equal. arXiv: 2410.01748, 2024. <https://arxiv.org/abs/2410.01748>, Jan. 2026.
- [210] Gao B, Song F, Yang Z *et al.* Omni-MATH: A universal olympiad level mathematic benchmark for large language models. arXiv: 2410.07985, 2024. <https://arxiv.org/abs/2410.07985>, Jan. 2026.
- [211] Wang Y, Li H, Zhang X, Wu J, Liu X, Hu W, Guo Z, Huang Y, Xin Y, Yang Y, Su J, Chen Q, Li S. EpiCoder: Encompassing diversity and complexity in code generation. In *Proc. the 42nd International Conference on Learning Representations*, Jul. 2025.
- [212] Xu Z, Liu Y, Yin Y, Zhou M, Poovendran R. KodCode: A diverse, challenging, and verifiable synthetic dataset for coding. In *Proc. the 2025 Findings of the Association for Computational Linguistics*, Jul. 2025, pp.6980–7008. DOI: [10.18653/v1/2025.findings-acl.365](https://doi.org/10.18653/v1/2025.findings-acl.365).
- [213] Jimenez C E, Yang J, Wettig A, Yao S, Pei K, Press O, Narasimhan K R. SWE-bench: Can language models resolve real-world GitHub issues? In *Proc. the 12th International Conference on Learning Representations*, May 2024.
- [214] Aleithan R, Xue H, Mohajer M M, Nnorom E, Uddin G, Wang S. SWE-bench+: Enhanced coding benchmark for LLMs. arXiv: 2410.06992, 2024. <https://arxiv.org/abs/2410.06992>, Jan. 2026.
- [215] Liang S, Garg S, Moghaddam R Z. The SWE-bench illusion: When state-of-the-art LLMs remember instead of reason. arXiv: 2506.12286, 2025. <https://arxiv.org/abs/2506.12286>, Jan. 2026.
- [216] Lu P, Mishra S, Xia T, Qiu L, Chang K W, Zhu S C, Tafjord O, Clark P, Kalyan A. Learn to explain: Multi-modal reasoning via thought chains for science question answering. In *Proc. the 36th International Conference on Neural Information Processing Systems*, Nov. 28-Dec. 9, 2022, pp.2507–2521. DOI: [10.5555/3600270.3600452](https://doi.org/10.5555/3600270.3600452).
- [217] Qin Y, Liang S, Ye Y *et al.* ToolLLM: Facilitating large language models to master 16000+ real-world APIs. In *Proc. the 12th International Conference on Learning Representations*, May 2024.
- [218] Wu J, Yin W, Jiang Y, Wang Z, Xi Z, Fang R, Zhang L, He Y, Zhou D, Xie P, Huang F. WebWalker: Benchmarking LLMs in web traversal. In *Proc. the 63rd Annual Meeting of the Association for Computational Linguistics*, Jul. 2025, pp.10290–10305. DOI: [10.18653/v1/2025.acl-long.508](https://doi.org/10.18653/v1/2025.acl-long.508).



**Hao Liang** is currently a Ph.D. candidate at Peking University (PKU), Beijing. His research focuses on data-centric machine learning, with particular emphasis on data preparation systems and algorithms for large language models (LLMs).



**Zhen Hao Wong** is currently a master student at Peking University (PKU), Beijing. His research focuses on data-centric machine learning and large language models (LLMs).



**Rui-Tong Liu** is currently a master student at Peking University (PKU), Beijing. Her research focuses on data-centric machine learning and multi-modal large language models (MLLMs).



**Yu-Han Wang** is currently a master student at Peking University (PKU), Beijing. His research focuses on data-centric machine learning and LLM-based agents.



**Mei-Yi Qiang** is currently a master student at Peking University (PKU), Beijing. Her research focuses on data-centric machine learning and data-model interactive training systems.



**Zheng-Yang Zhao** is currently a Ph.D. student at Peking University (PKU), Beijing. His research focuses on data-centric machine learning and LLM-based agents.



**Cheng-Yu Shen** is currently a master student at Peking University (PKU), Beijing. His research focuses on data-centric machine learning and LLM evaluation systems.



**Cong-Hui He** is a young leading scientist at the Shanghai AI Laboratory and an adjunct doctoral supervisor at the School of Artificial Intelligence, Shanghai Jiao Tong University, Shanghai. He received his Ph.D. degree from Tsinghua University, Beijing. His research focuses on data-centric artificial intelligence and high-performance computing, with significant impact on both academia and industry. His honors include the Gordon Bell Prize, the ACL Best Theme Paper Award, and the WAIC Yunfan Award.



**Wen-Tao Zhang** is an assistant professor in the Center for Machine Learning Research at Peking University (PKU), Beijing. His research focuses on graph machine learning (ML), data-centric ML, ML systems and AI4Science. His research works have been powering several billion-scale applications in Tencent, and some of them have been recognized by multiple best paper awards, including the Best Paper Runner Up Award at APWeb-WAIM 2023, and the Best Student Paper Award at WWW 2022.



**Bin Cui** is a professor in the School of Computer Science at Peking University, Beijing. He is a fellow of IEEE (2024) and Cheung Kong distinguished Professor (2016). His research interests include database system, big data management and analytics, ML/DL system. Prof. Cui has published more than 100 research papers and has served in the Technical Program Committee of various international conferences, including SIGMOD, VLDB, ICDE, and KDD. He was a recipient of VLDB 2022 SDS Best paper award, WWW 2022 Best student paper award, and NSR 2016 best paper award. He was awarded the Second Prize of the Natural Science Award from the Ministry of Education (MOE), China in 2014.