



HUGGINGBREW: Brewing High-Quality Mid-Training Data from the Hugging Face Community Datasets

Han Zhang^{1,2,3*}, Ningqi Chen^{1,2}, Xiao Liu³, Zhenghao Lin³,
Yeyun Gong^{3,†}, Yan Lu³, Hanfang Yang^{1,2,†}

¹Center for Applied Statistics, Renmin University of China

²School of Statistics, Renmin University of China

³Microsoft Research Asia

Abstract

Data curation for large language models (LLMs) has been extensively studied for both the pre-training and post-training stages, yet the intermediate *mid-training* phase—bridging general pretraining and task-specific fine-tuning—remains critically underexplored. Most existing mid-training data for state-of-the-art models remains proprietary. Meanwhile, the Hugging Face platform hosts a huge amount of community-contributed datasets, yet no systematic effort has been made to refine this rich ecosystem into a unified high-quality corpus. We present **HUGGINGBREW**, a large-scale mid-training corpus constructed through a principled, multi-step curation pipeline applied to the massive publicly accessible Hugging Face datasets. The pipeline operates at two complementary granularities: *dataset-level* operations act on each dataset independently, while *corpus-level* operations are applied jointly across all retained corpus. We construct three progressively curated variants—DRAUGHT, CRAFT, and RESERVE—containing 1.6T, 450B, and 40B tokens, respectively. Continual pretraining experiments under a budget of up to 100B tokens show that HUGGINGBREW series yields substantial improvements over baselines across mathematical reasoning, general knowledge, and code generation, validating the untapped value of community-contributed data for mid-training.

1 Introduction

The rapid advancement of Large Language Models (LLMs) (OpenAI, 2025; Meta AI, 2025; Yang et al., 2025; DeepSeek-AI, 2026) has been largely driven by the scaling and refinement of training data. Existing data curation efforts have predominantly concentrated on two stages. On one end,

* Work done during internships at Microsoft Research Asia.

† Corresponding to Hanfang Yang (hyang@ruc.edu.cn) and Yeyun Gong (yegong@microsoft.com). Hanfang Yang is the first corresponding author.

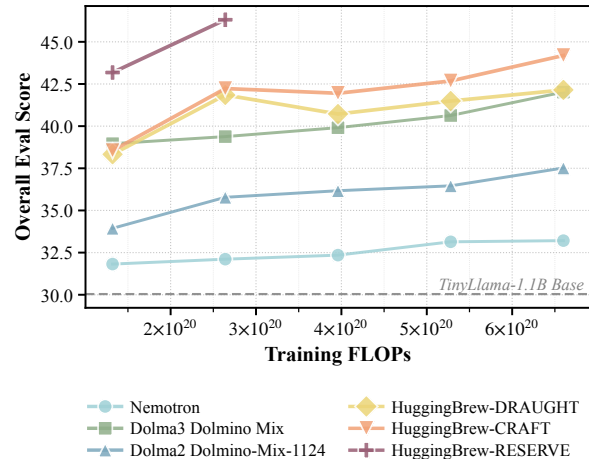


Figure 1: Downstream evaluation performance of the HUGGINGBREW series under the same training budget.

considerable work has focused on building massive **pre-training** corpora from large-scale web crawls, including C4 (Raffel et al., 2020) and The Pile (Gao et al., 2020), with more recent efforts such as FineWeb (Penedo et al., 2024), DataComp-LM (DCLM) (Li et al., 2024a), and Nemotron-CC (Su et al., 2024) pushing the frontier of fine-grained quality filtering and data enrichment. On the other end, substantial attention has been devoted to **post-training** (Kumar et al., 2025; Tie et al., 2025; Wang et al., 2024), where supervised fine-tuning (SFT) and reinforcement learning (RL) are used to instill targeted capabilities (Zhang et al., 2025b), strengthen instruction following abilities and align model outputs with human preferences.

However, between the large-scale pre-training and the targeted refinement of post-training lies a critical intermediate phase: **mid-training** (Mo et al., 2025; OpenAI, 2024). Mid-training serves as a crucial bridge that adapts a generalist base model to a more curated data mixture before final alignment, and has emerged as an effective strategy for improving generalization and training efficiency in subsequent stages (Abdin et al., 2024; Grattafiori et al., 2024; Yang et al., 2025). Despite its growing

importance, most mid-training mixtures developed by the industry are proprietary, leaving a significant gap in the open-source ecosystem for a systematic, large-scale mid-training corpus.

Platforms such as Hugging Face Datasets¹ host tons of community-uploaded datasets spanning diverse domains, tasks, and languages. Although their overall quality is highly uneven, these datasets are typically stored in structured tabular formats with relatively standardized schemas, in contrast to noisy web crawls that require trivial HTML-to-text extraction. This makes them a promising yet largely untapped source for mid-training, a stage that demands data combining adequate scale with high content quality. To date, however, existing efforts in mid-training—including fully open-source projects such as the OLMo series (OLMo et al., 2024, 2025)—rely on careful selection and refinement of a small number of datasets, and no systematic effort has yet been made to harvest, refine, and unify this ecosystem into a high-quality corpus directly usable for LLM training.

To bridge this gap, we present 🧑🔧 **HUGGINGBREW**, the first systematic, large-scale curation pipeline applied to community Hugging Face datasets. Starting from a massive initial pool of publicly crawled datasets from the Hugging Face Hub, our pipeline proceeds through **nine** carefully designed steps under two complementary granularities: *dataset-level* operations on each dataset and *corpus-level* operations on their union. The former mainly performs filtering, duplicate detection, and quality scoring based on characteristics from reservoir samples, while the latter addresses cross-dataset deduplication, decontamination and compliance processing. From this pipeline, we progressively construct three variants of increasing refinement—DRAUGHT, CRAFT, and RESERVE—comprising 1.6T, 450B, and 40B tokens of high-quality mid-training data, respectively. Continual pretraining experiments under a budget of up to 100B tokens show that each key stage of HUGGINGBREW generally yields progressive improvements, and that HUGGINGBREW achieves substantial gains over other open-source mid-training corpora on benchmarks spanning mathematical reasoning, general knowledge, and code generation.

In summary, our key contributions are as follows: (i) We introduce a systematic, **nine-step data curation pipeline** specifically designed to handle the

massive Hugging Face datasets. (ii) We construct the **HUGGINGBREW** series (HUGGINGBREW-DRAUGHT/CRAFT/RESERVE), comprising mid-training corpora at different scales and levels of refinement to support diverse training budgets and quality requirements. (iii) We conduct extensive continual pretraining experiments to validate the effectiveness of HUGGINGBREW, **demonstrating significant improvements** over baselines across a wide range of downstream benchmarks.

2 Related Work

Mid-Training. Recent advances have highlighted the critical role of the **mid-training** phase, in which refined data are leveraged under an annealing learning-rate schedule (Jin et al., 2023). Although the terminology varies, the paradigm of multi-stage base model training has been widely adopted: OpenAI uses the term to denote the stage that injects new domain-specific knowledge into foundation models (OpenAI, 2024); Llama 3 (Grattafiori et al., 2024) realizes it as an annealing stage that shifts the data distribution toward high-quality sources; DeepSeek-V3 (DeepSeek-AI, 2024) and Phi-4 (Abdin et al., 2024) show that careful data composition at this stage yields substantial downstream gains. The OLMo series (OLMo et al., 2024, 2025), which open-sources the full training lifecycle, additionally releases its curated mid-training data. However, all existing mid-training mixtures—whether proprietary or open—are typically assembled from a handful of handpicked sources, leaving community-contributed datasets largely unexplored for this critical phase.

Community Data Aggregation. Several initiatives have aggregated open-source community contributions. The ROOTS (Laurençon et al., 2022) assembled a 1.6 TB multilingual dataset through coordinated community effort, The Stack (Kocetkov et al., 2023) harvested permissively licensed code from GitHub, and RedPajama (Weber et al., 2024) incorporated question-answer data from community forums such as StackExchange. To our knowledge, no prior work has systematically harvested the full ecosystem of publicly available Hugging Face datasets. This naturally raises a compelling question: given that the Hugging Face Hub serves as a large-scale data hosting platform where each dataset is inherently stored as structured tabular data, *can we systematically treat the entire collection of Hugging Face datasets as heterogeneous*

¹<https://huggingface.co/datasets>

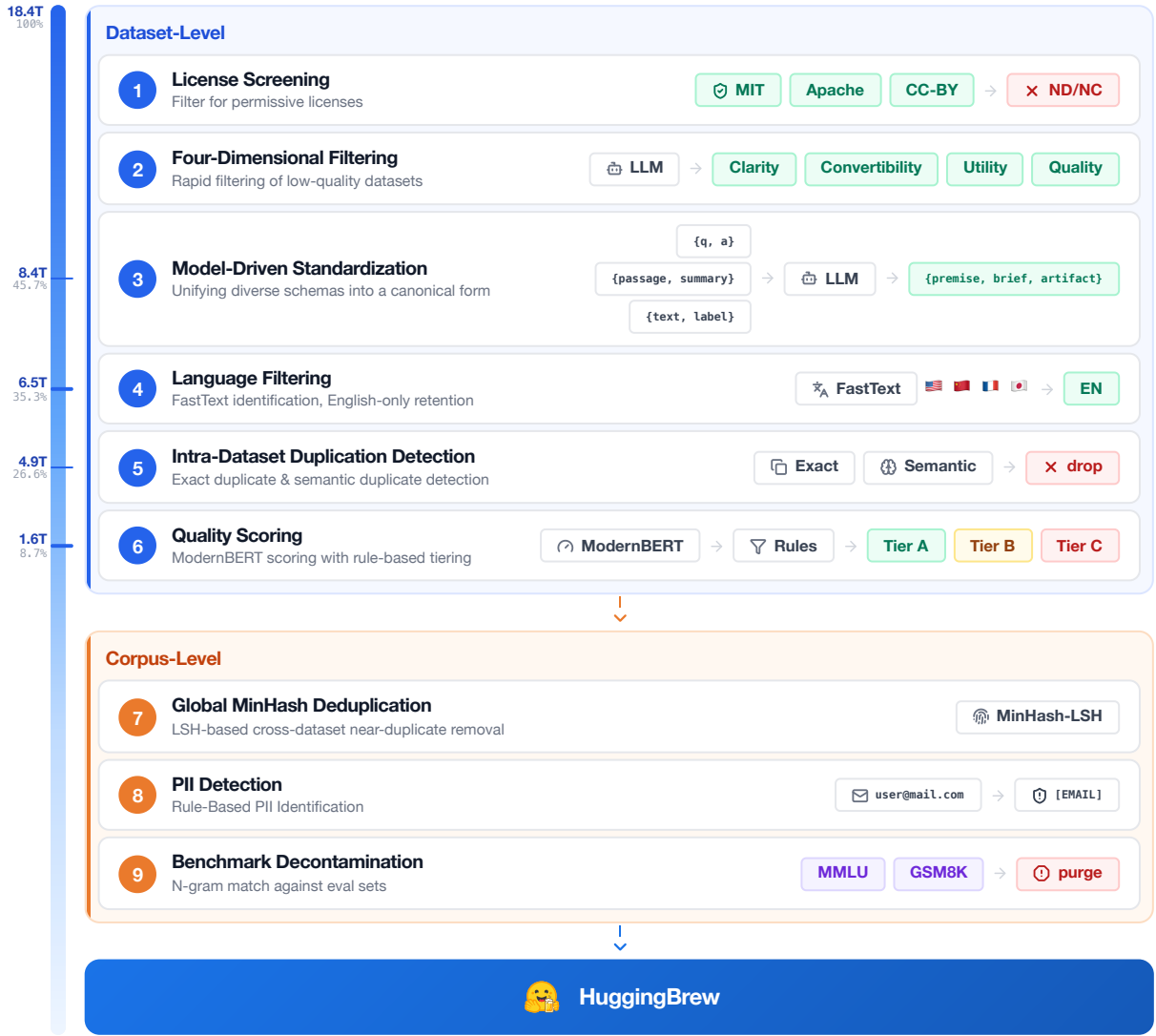


Figure 2: Overview of the nine-step data curation pipeline.

tables, unify them through a principled curation pipeline, and ultimately organize them into a coherent corpus? HUGGINGBREW provides an affirmative answer.

3 Methodology

We present the HUGGINGBREW data curation pipeline, which transforms a large-scale crawl of publicly available Hugging Face datasets into a high-quality, ready-to-use training corpus. Motivated by the observation that Hugging Face datasets are highly heterogeneous across sources but follow specific schema within each individual dataset, the pipeline is organized into two complementary phases: *dataset-level processing* (§3.2) operates on each dataset independently, while *corpus-level processing* (§3.3) operates on the union of all retained datasets. We first introduce the notation used throughout this section.

3.1 Preliminaries

Our pipeline begins with a large-scale crawl of publicly available datasets hosted on Hugging Face. Each resource on the platform is uniquely identified by a hierarchical path of the form organization/dataset, with corresponding splits (e.g., train, validation, test). In this work, we treat each split as the finest granularity at which data is stored and accessed, *i.e.*, each split is regarded as an independent dataset. Further details on data crawling and basic preprocessing are detailed in Appendix A.

Dataset Representation. Let $\mathcal{D} = \{D_i\}_{i=1}^N$ denote the full collection of N crawled datasets. Each dataset D_i is a structured table with n_i rows and m_i typed feature columns:

$$D_i = \{\mathbf{x}_j\}_{j=1}^{n_i}, \quad \mathbf{x}_j = (x_{j,k})_{k=1}^{m_i}, \quad (1)$$

where $x_{j,k}$ denotes the column c_k 's value at row j .

3.2 Dataset-Level Processing

While the column schemas are highly heterogeneous across datasets in both data type and semantics, samples within an individual dataset generally share a common format, task objective, and schema. We exploit this property by performing the first phase of curation at the dataset level: for each $D_i \in \mathcal{D}$, we produce either a keep/discard decision or a transformed version D'_i , based on global properties or on a small sampled subset $\tilde{D}_i \subset D_i$ with $|\tilde{D}_i| \ll |D_i|$. Prior works have shown that representative characteristics extracted from such samples can reliably approximate dataset-wide properties (Sui et al., 2024; Zhang et al., 2025a), making it feasible to process the massive volume of crawled datasets at scale. This phase comprises six steps that progressively filter, transform, and assess datasets before they are merged into the unified corpus. The implementation details and hyperparameter settings are provided in Appendix B.

Step 1: License Screening. We first perform systematic license screening to ensure the legal compliance of all retained datasets. Specifically, we identify and remove datasets whose licenses explicitly prohibit the modification or redistribution of derivative works (e.g., CC-BY-ND, CC-BY-NC-ND). A complete list of the permissive licenses retained in our final corpus is provided in Appendix C.

Step 2: Four-Dimensional Filtering. Despite hosting a vast number of datasets, the Hugging Face ecosystem exhibits highly uneven data quality, with a substantial portion consisting of low-quality user uploads. Unlike raw web crawls that require extensive HTML extraction, these datasets are already well-structured in format, yet they may be semantically incoherent and lack systematic quality control. To efficiently filter out such datasets at an early stage, we design a **four-dimensional filtering** mechanism that leverages an LLM to evaluate each dataset based on sampled rows along four complementary dimensions. The output includes three binary gates, one graded score, and a descriptive summary. The four dimensions are:

- **Clarity** (binary): Retains datasets with sufficient information, discarding those with opaque or missing metadata.
- **Convertibility** (binary): Ensures the schema can be programmatically mapped to our canonical formats, excluding purely non-textual (e.g., embeddings arrays) or irreducibly nested data.

- **Utility** (binary): Filters out low-information content in favor of datasets with substantive linguistic, reasoning, or educational value.
- **Quality** (0–5 scale): Provides a comprehensive evaluation of the dataset’s overall quality.

A dataset is retained only if it passes all three binary gates and receives a quality score above 3, ensuring fundamental data quality for all subsequent stages.

Step 3: Model-Driven Format Standardization.

The retained datasets still differ in schema semantics, metadata fields, and storage formats, making it infeasible to design a single processing pipeline for all of them. To address this, we propose a **model-driven format standardization** approach. Since all rows within a given dataset share the same feature schema, we provide an LLM with sampled rows along with the dataset summary and instruct it to generate a transformation function that maps each row into three canonical content fields: premise, brief, and artifact. Specifically, premise captures background information and context; brief contains the primary content to be processed; and artifact records the expected follow-up or response. The function also stores fields not mapped to training content in an append metadata field, which is preserved for provenance but excluded from downstream corpus processing and model training. The generated functions are automatically validated on a sampled subset to ensure correctness, and then applied to the entire dataset to produce a standardized version ready for downstream processing. Concrete examples are provided in Appendix E.

Step 4: Language Identification. Although Hugging Face provides language-related tags for hosted datasets, we independently perform language identification at the dataset level. We adopt the FT176 language identification (LID) classifier from FineWeb (Penedo et al., 2024), sampling a subset of rows from each dataset and applying the LID model to each sampled row. Predictions exceeding a predefined confidence threshold are retained, and the most frequently predicted language among all confident samples is assigned as the dataset-level language label. Datasets for which no dominant language can be determined are designated as *Unknown*. We retain only English-related datasets in our pipeline. The resulting language distribution is summarized in Figure 6.

Step 5: Intra-Dataset Duplication Detection.

Large-scale deduplication—including both exact and semantic methods—has been widely adopted in data processing pipelines (Lee et al., 2022; Abbas et al., 2023; Penedo et al., 2024). However, existing approaches typically operate globally across the entire corpus, treating all documents as an undifferentiated pool. This strategy is not directly suitable for Hugging Face datasets: repetitive datasets with high intra-dataset redundancy must be identified and addressed before corpus-level processing. Rather than performing expensive sample-level deduplication within every dataset, we design an efficient **dataset-level duplication detection** mechanism that identifies and discards highly redundant datasets before they enter the corpus.

Concretely, for each dataset we compute two complementary metrics: (1) the *exact duplicate ratio*, measured via hash encoding across the entire dataset, and (2) the *average semantic similarity*, computed as the mean pairwise cosine similarity over dense embeddings for sampled subsets. Datasets exceeding a predefined threshold on either metric are deemed excessively redundant and discarded entirely.

Step 6: Fine-Grained Quality Scoring. A variety of scorers have been proposed to assess text quality, with recent work (Zhuang et al., 2025; Cai et al., 2025; Liang et al., 2025) further organizing and ensembling them for more robust evaluation. To efficiently assess the large volume of remaining datasets, we distill the outputs of multiple orthogonal scorers into a unified model and derive filtering rules at varying levels of strictness for dataset-level selection. Drawing on the systematic comparison in OpenDataArena (Cai et al., 2025), we adopt the following five scoring dimensions, which jointly cover complementary aspects of data quality:

- **Deita Quality:** The Deita quality scorer (Liu et al., 2024) evaluates the overall writing quality of each artifact, assessing linguistic fluency, structural clarity, and informational density.
- **Reward Model:** Skywork-Reward (Liu et al., 2025) provides reliable signals for estimating human preference alignment, reflecting the degree to which an artifact matches human expectations in helpfulness and coherence.
- **Instruction-Following Difficulty (IFD):** The IFD score (Li et al., 2024b) measures the alignment between a brief and its corresponding artifact, providing a cost-effective signal for

detecting mismatched pairs.

- **Correctness:** An LLM-as-Judge scorer targets factual accuracy and logical soundness.
- **Relevance:** An LLM-as-Judge scorer assesses the topical relevance of each artifact to its brief. Unlike statistical measures such as IFD, this scorer performs direct semantic evaluation and is robust to variations in text length.

To enable efficient scoring at scale, we distill the above scorers into a unified model using ModernBERT (Warner et al., 2025). The [CLS] representation from the encoder is passed through a shared projection MLP, then fed into five independent linear heads, each predicting one of the five quality dimensions. The distillation model is trained on a mixed dataset of 100K samples annotated by the respective scorers, with all scores normalized via z -score standardization on the training set.

Based on the five-dimensional scores, we design two sets of data selection rules with different levels of strictness to holistically assess each dataset. We employ a sampling strategy to characterize the quality distribution within each dataset and filter based on statistical properties of the sampled scores, including concentration, variance, and mean. Datasets are ultimately categorized into three quality tiers—A, B, and C. The detailed selection criteria are summarized in Table 8 and Table 9.

3.3 Corpus-Level Processing

After dataset-level processing, the M retained Tier-A datasets ($M \leq N$) are merged into a unified corpus $\mathcal{C} = \bigcup_i D_i'$. Corpus-level processing then operates directly on $\mathcal{C}$ to produce a refined corpus $\mathcal{C}' \subseteq \mathcal{C}$, addressing cross-dataset redundancy and contamination with personal information and evaluation benchmarks.

Step 7: Global MinHash Deduplication. While Step 5 filters out datasets with excessive internal redundancy, substantial cross-dataset duplication persists within our data pool. Different users may upload duplicate copies or variants of the same dataset, making corpus-level deduplication essential. To this end, we perform global fuzzy deduplication over the entire corpus using MinHash with Locality-Sensitive Hashing (LSH). Following the overall MinHash-LSH design of FineWeb (Penedo et al., 2024), we retain a similar approximate Jaccard threshold of 0.75 but use 126 permutations divided into 14 LSH buckets (nine hashes per bucket). As shown in Table 3, this configuration strikes a

Table 1: Evaluation results on downstream benchmarks after continual pre-training with different data configurations. Each row labelled **S1** $\Rightarrow$ **Sk** denotes the corpus obtained by executing all curation stages from S1 through Sk. **Bold** indicates the best result per benchmark within each model group; underline indicates the second best.

Configuration	General Knowledge							Math Reasoning		Code	Avg
	MMLU	HellaSwag	OBQA	WinoGrande	ARC-C	ARC-E	PIQA	GSM8K	MATH	HumanEval	
TinyLlama-1.1B											
Baselines											
Base	27.07	53.80	27.00	50.67	29.15	39.33	64.31	2.35	1.22	5.49	30.04
Nemotron-CC-HQ	28.80	60.51	27.40	<u>57.93</u>	30.51	48.15	74.48	2.50	1.72	0.00	33.20
Dolma 2 Mid-Train	37.05	<u>57.05</u>	30.20	58.56	33.56	40.56	72.20	33.43	5.06	4.88	37.26
Dolma 3 Mid-Train	47.22	54.73	35.00	57.22	31.53	46.74	71.60	39.20	15.68	21.34	42.03
HUGGINGBREW Pipeline											
S1 ⇒ S3	36.92	53.00	35.40	56.35	33.22	42.50	69.15	40.03	12.80	18.90	39.83
S1 ⇒ S4	42.97	54.64	<u>41.20</u>	56.64	31.19	40.74	72.36	28.81	17.66	23.17	40.94
S1 ⇒ S5	41.96	56.78	41.00	56.04	32.54	41.80	72.63	45.19	19.54	15.85	42.33
HuggingBrew-DRAUGHT	41.46	56.86	32.40	56.27	<u>34.24</u>	44.44	71.22	51.78	19.74	15.24	42.37
HuggingBrew-CRAFT	40.07	56.69	39.00	57.38	<u>34.24</u>	54.85	<u>72.69</u>	<u>50.72</u>	25.92	9.15	<u>44.07</u>
HuggingBrew-RESERVE	<u>44.84</u>	55.06	51.60	56.43	36.27	<u>53.79</u>	71.06	49.05	<u>21.48</u>	23.17	46.28
OpenELM-3B											
Baselines											
Base	26.95	64.75	21.80	58.48	32.20	44.97	74.37	4.02	2.14	7.32	33.70
Nemotron-CC-HQ	29.71	67.60	25.20	59.43	32.88	46.91	<u>75.19</u>	4.70	2.10	0.00	34.37
Dolma 2 Mid-Train	42.36	65.32	35.20	59.27	<u>34.58</u>	39.86	73.61	37.07	5.14	10.37	40.28
Dolma 3 Mid-Train	<u>44.25</u>	68.25	<u>46.00</u>	59.43	33.90	44.80	75.73	16.00	2.84	16.46	40.77
HUGGINGBREW Pipeline											
S1 ⇒ S3	40.10	65.77	32.40	61.72	<u>34.58</u>	<u>48.15</u>	73.83	37.53	10.38	18.90	42.34
S1 ⇒ S4	44.42	66.12	40.40	60.06	33.56	40.56	74.27	37.60	14.70	15.85	42.75
S1 ⇒ S5	43.21	65.69	35.20	60.54	35.59	41.62	74.37	43.97	12.30	20.73	43.32
HuggingBrew-DRAUGHT	43.83	60.00	36.00	58.09	35.59	41.27	71.49	<u>47.69</u>	<u>16.84</u>	<u>21.95</u>	43.28
HuggingBrew-CRAFT	42.37	<u>68.14</u>	40.80	<u>61.17</u>	32.54	43.03	74.70	44.96	11.24	23.17	<u>44.21</u>
HuggingBrew-RESERVE	42.63	63.33	48.40	60.38	34.24	51.85	74.16	52.62	17.16	14.02	45.88

favorable balance between document retention and downstream performance.

Step 8 & 9: PII Detection & Benchmark Decontamination. We first apply PII detection over the entire corpus to identify samples containing sensitive personal information, such as names, addresses, and phone numbers, thereby ensuring privacy compliance. To prevent the numerous benchmarks and related datasets hosted on Hugging Face from leaking into the training data, the final stage of our pipeline performs n -gram-based contamination detection and filtering. Specifically, following DCLM (Li et al., 2024a), we decontaminate the corpus against all 72 evaluation subsets used in our work, which are detailed in Appendix H. For other benchmarks, we do not apply targeted decontamination; instead, we preserve complete source attribution and metadata in the curated corpus, enabling additional auditing or filtering as needed.

All corpus-level operations are implemented and modified based on the `datatrove` library.²

²<https://github.com/huggingface/datatrove>

3.4 The HUGGINGBREW Series

To accommodate diverse experimental and downstream requirements, we construct three HUGGINGBREW variants, each comprising training data at a different scale and quality level:

- **HUGGINGBREW-DRAUGHT:** the full corpus obtained after the fine-grained quality scoring step, retaining all Tier-A subsets. This variant maximizes data volume and knowledge coverage, without further cross-dataset deduplication or decontamination. Despite skipping these corpus-level steps, it still enforces strict dataset-level quality control and offers the richest collection of community-contributed knowledge.
- **HUGGINGBREW-CRAFT:** a standard subset after the whole nine-step pipeline, striking a favorable balance between data quality and scale.
- **HUGGINGBREW-RESERVE:** a highly selective variant with the strictest filtering criteria of Step 6, followed by the same corpus-level deduplication and decontamination procedures. It achieves the highest data quality at the cost

of a reduced scale, making it particularly suitable for training stages with demanding quality requirements.

4 Experiments

To validate the effectiveness of our data curation pipeline, we conduct a series of mid-training experiments and evaluate on a comprehensive suite of downstream benchmarks spanning general knowledge, mathematical reasoning and code generation.

4.1 Experimental Setup

Mid-Training Setup. We use TinyLlama-1.1B (Zhang et al., 2024) and OpenELM-3B (Mehta et al., 2024) as model backbones for the main experiments. To assess the impact of each key curation stage, we treat each successive stage as an independent data pool and sample up to 100B tokens for continual pretraining (averaged over two independently drawn samples). If a pool contains fewer than 100B tokens, we use the entire pool. Full experimental details, including learning rate schedules and hardware configurations, are provided in Appendix B. We denote each data pool by the range of curation stages it has undergone:

- **S1 $\Rightarrow$ S3:** the standardized corpus produced after format standardization (S3), serving as the starting point of the ablation.
- **S1 $\Rightarrow$ S4:** further restricted to English-dominant datasets via language identification (S4).
- **S1 $\Rightarrow$ S5:** removes highly redundant datasets via intra-dataset duplication detection (S5).
- **HUGGINGBREW series:** three variants DRAUGHT, CRAFT, and RESERVE, obtained at different stages and filtering levels of the curation pipeline, as described in §3.4.

Baselines. We compare HUGGINGBREW against the following baselines:

- **Nemotron-CC-HQ:** The high-quality subset of Nemotron-CC (Su et al., 2024), filtered by its model-based quality classifier.
- **Dolma 3 Mid-Train³:** The OLMo 3 (OLMo et al., 2025) mid-training data mixture.
- **Dolma 2 Mid-Train⁴:** The OLMo 2 (OLMo et al., 2024) mid-training mixture.

To match the training scale, we randomly sample 100B tokens from the HQ subset of Nemotron-CC

for continual pretraining. For the Dolma series, we follow the official 100B-token mixing ratio for Dolma 2 and directly adopt the officially released mid-training mixture for Dolma 3.

Evaluation Benchmarks. We evaluate across three capability dimensions: (1) *math reasoning*: GSM8K (Cobbe et al., 2021) and MATH (Hendrycks et al., 2021b); (2) *code*: HumanEval (Chen et al., 2021); and (3) *general knowledge*: MMLU (Hendrycks et al., 2021a), HellaSwag (Zellers et al., 2019), OpenBookQA (Mihaylov et al., 2018), WinoGrande (Sakaguchi et al., 2020), ARC-Challenge/Easy (Clark et al., 2018), and PIQA (Bisk et al., 2020). All evaluations are conducted using the OpenCompass (Contributors, 2023) framework.

4.2 Main Results

Table 1 reports the final evaluation results across all benchmarks, from which we draw several key observations. Compared with Nemotron-CC-HQ and the Dolma series, our pipeline yields substantial improvements on most benchmarks, with the largest gains observed in mathematical reasoning. Notably, these baselines rely heavily on synthetic generation and augmentation over the raw data, whereas HUGGINGBREW preserves the original content of community-contributed datasets, highlighting the untapped potential of Hugging Face community data. Across both model backbones, average performance generally improves as the curation process becomes more selective, although individual benchmarks and intermediate configurations exhibit non-monotonic variation. In particular, HUGGINGBREW-RESERVE achieves the highest average score for both TinyLlama-1.1B (46.28) and OpenELM-3B (45.88), supporting the overall effectiveness of our curation design.

4.3 Further Analysis

Analysis on Sampling Sufficiency. During HUGGINGBREW pipeline, we employ reservoir sampling to enable efficient dataset-level screening. To examine the sufficiency of this sampling strategy, we conduct a controlled study on 210 sampled datasets, stratified by row count into Small ($< 5K$, 100 files), Medium (5K–50K, 80 files), and Large ($> 50K$, 30 files). For each file we draw $K \in \{50, 100, 200, 500, 2000\}$ rows and re-run the three pipeline steps (Steps 4–6)—language identification, intra-dataset semantic similarity, and fine-

³https://huggingface.co/datasets/allenai/dolma3_dolmino_mix-100B-1125

⁴<https://huggingface.co/datasets/allenai/dolmino-mix-1124>

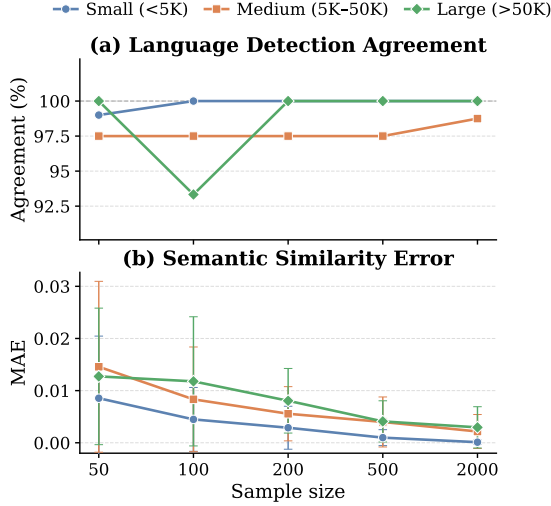


Figure 3: Ablation results comparing different sampling strategies for language identification and intra-dataset semantic similarity.

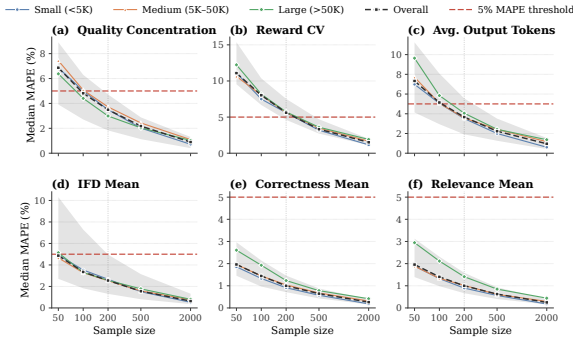


Figure 4: Mean absolute percentage error (MAPE) of the different scoring dimensions. The red dashed line marks the 5% error level.

grained quality scoring. Figures 3 and 4 summarize the findings. Language detection is highly sample-efficient: overall agreement exceeds 97% at $K = 200$ across all dataset tiers. For semantic similarity, the mean absolute error (MAE) decreases monotonically with sample size across all three tiers, yet remains consistently low even at the smallest sample sizes. For the quality scoring dimensions, we report the mean absolute percentage error (MAPE) to jointly assess different scorers across sample sizes. Overall, MAPE decreases monotonically with K across all dimensions and drops below 5% relative error already at $K = 200$, indicating that reservoir sampling reliably estimates data quality with a small fraction of the full data.

Analysis on Quality Scorer. We compare our quality scorer, paired with two filtering rule sets of different strictness, against two representative scorer-based filtering schemes: the FineWeb-

Table 2: Analysis on the quality scorer. “Avg” denotes the average performance over all benchmarks. All variants share an identical pipeline outside of Step 6.

Quality scorer	Avg
FineWeb-Edu (Penedo et al., 2024)	41.26
Meta-rater PRRC (Zhuang et al., 2025)	40.77
Ours (Standard)	42.37
Ours (Stringent)	43.75

Table 3: Analysis on MinHash configuration. “Threshold” denotes the approximate Jaccard similarity threshold, and “Avg” denotes the average performance over all benchmarks. The bold row corresponds to the setting we adopted.

# Perm.	# Buckets	Threshold	Retention	Avg
24	8	~ 0.50	34.4%	41.98
126	14	~ 0.75	56.3%	44.18

Edu (Penedo et al., 2024) classifier and the four-dimensional PRRC scorer (Professionalism, Readability, Reasoning, Cleanliness) from Meta-rater (Zhuang et al., 2025). As shown in Table 2, our scorer outperforms both alternatives on average by a clear margin (+2.49 over FineWeb-Edu and +2.98 over Meta-rater PRRC), indicating that it provides a more effective assessment of data quality for mid-training than existing scorers.

Analysis on MinHash Configuration. To assess the appropriateness of our MinHash configuration, we additionally evaluate a more aggressive setting with only 24 permutations (8 bands of 3 hashes each), which yields a lower approximate Jaccard threshold and therefore identifies more pairs as near-duplicates. Table 3 illustrates the approximate Jaccard similarity above which a pair is treated as a near-duplicate, the resulting document retention rate, and the downstream performance. Our adopted setting substantially outperforms the aggressive one (+2.20 Avg), indicating that the more aggressive, lower-threshold configuration removes too much informative content.

Duplicate-Detection Landscape. We analyze the redundancy patterns observed during the Step-5 intra-dataset duplicate detection. As shown in Figure 5, the distribution of exact-duplicate rates is heavily right-skewed: the vast majority of datasets exhibit low exact-duplicate rates, while a small but non-negligible tail contains highly repetitive corpora. Accordingly, we adopt a threshold of $\theta_{\text{exact}} = 0.30$ to filter out datasets with excessive

verbatim repetition. For semantic similarity, we observe a pronounced peak near the upper end of the distribution, indicating that some datasets contain extensive near-paraphrastic content despite falling below the exact-duplicate-rate threshold; we therefore apply a stricter threshold of $\theta_{\text{sem}} = 0.95$ to remove such cases. Together, these two thresholds effectively prune highly redundant datasets and improve the overall quality and diversity of the curated corpus.

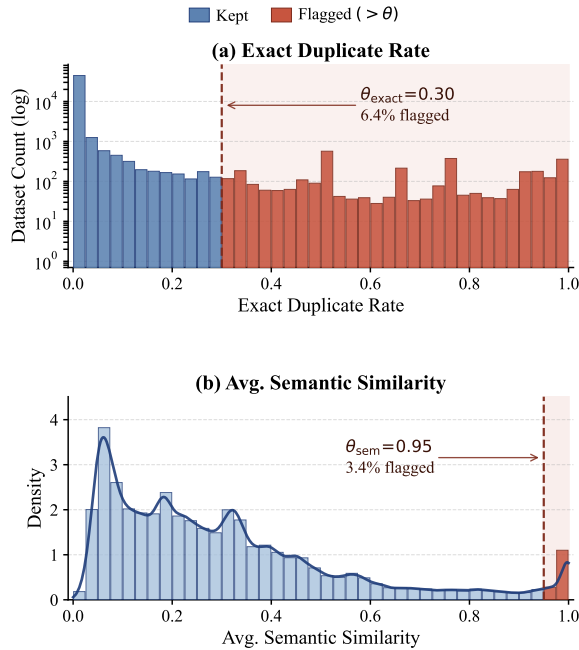


Figure 5: Duplicate-detection landscape across Hugging Face datasets.

5 Conclusion

We present HUGGINGBREW, a large scale mid-training corpus curated from the diverse and rapidly growing Hugging Face community datasets. Extensive experiments and analysis demonstrate that our pipeline effectively unifies heterogeneous sources into a coherent corpus which improves model performance across a wide range of benchmarks. We hope HUGGINGBREW serves as a practical solution for harnessing the wealth of knowledge embedded in Hugging Face datasets, and inspires further research on data-centric model improvement.

Limitations

While we conduct rigorous license filtering based on the metadata provided by dataset uploaders on Hugging Face, we acknowledge that the declared licenses may not always reflect the true intellectual property status of the underlying content. In particular, some datasets on Hugging Face are them-

selves derived from third-party sources (e.g., web scrapes of Reddit, Wikipedia, or proprietary forums) and re-uploaded under permissive licenses such as Apache-2.0 or MIT by their uploaders. In such cases, the uploader-assigned license may not carry legal authority over the original content, as the upstream rights holders never granted redistribution permission. This *transitive license* problem is a systemic challenge across the open-data ecosystem and is not unique to our work; similar concerns apply to widely used corpora such as RedPajama and The Pile. We mitigate this risk by (1) excluding datasets with unresolvable or restrictive licenses, (2) preserving full provenance metadata to support additional due diligence, and (3) retaining source-level identifiers so affected records can be traced and removed before any future public release. Nevertheless, establishing a fully verified chain of licensing from original content creators to the curated corpus remains infeasible at scale, and any future public release and downstream use must account for the requirements of the relevant legal jurisdictions.

Benchmark decontamination remains an open methodological challenge rather than a universally standardized preprocessing step. As noted by Nemotron-CC (Su et al., 2024), the field lacks consensus on an optimal procedure, and the effect of decontamination may depend on model scale and training horizon. DCLM (Li et al., 2024a) likewise emphasizes that the effect of evaluation-set overlap is not yet fully understood. Accordingly, we apply the n -gram-based protocol described in §3.3 to all 72 evaluation subsets used in this work. We admit this may fail to detect semantically equivalent or substantially paraphrased benchmark content (Yang et al., 2023). We preserve source provenance and metadata to facilitate further auditing, and leave the development of more robust semantic decontamination methods and broader benchmark coverage to future work and downstream users.

Acknowledgment

The authors would like to thank the reviewers for their constructive comments, which led to a significant improvement in this work. The research is supported by the Special Funds of the National Natural Science Foundation of China (Grant No. 72342010).

References

- Amro Abbas, Kushal Tirumala, Dániel Simig, Surya Ganguli, and Ari S. Morcos. 2023. SemDeDup: Data-efficient learning at web-scale through semantic deduplication. In *International Conference on Learning Representations*.
- Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar, Michael Harrison, Russell J. Hewett, Mojan Javaheripi, Piero Kauffmann, James R. Lee, Yin Tat Lee, Yuanzhi Li, Weishung Liu, Caio C. T. Mendes, Anh Nguyen, Eric Price, Gustavo de Rosa, Olli Saarikivi, and 8 others. 2024. Phi-4 technical report. *arXiv preprint arXiv:2412.08905*.
- Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. **PIQA: Reasoning about physical commonsense in natural language**. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7432–7439.
- Mengzhang Cai, Xin Gao, Yu Li, Honglin Lin, Zheng Liu, Zhuoshi Pan, Qizhi Pei, Xiaoran Shang, Mengyuan Sun, Zinan Tang, et al. 2025. OpenDataArena: A fair and open arena for benchmarking post-training dataset value. *arXiv preprint arXiv:2512.14051*.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.
- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.
- OpenCompass Contributors. 2023. OpenCompass: A universal evaluation platform for foundation models. <https://github.com/open-compass/opencompass>.
- DeepSeek-AI. 2024. DeepSeek-V3 technical report. *arXiv preprint arXiv:2412.19437*.
- DeepSeek-AI. 2026. DeepSeek-V4: Towards highly efficient million-token context intelligence. <https://huggingface.co/deepseek-ai/DeepSeek-V4-Pro>. Technical Report.
- Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Hrace He, Anish Thite, Noa Nabeshima, et al. 2020. The Pile: An 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, et al. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. **DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing**. In *International Conference on Learning Representations*.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021a. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021b. Measuring mathematical problem solving with the MATH dataset. In *Advances in Neural Information Processing Systems*, volume 34, pages 7294–7305.
- Hongpeng Jin, Wenqi Wei, Xuyu Wang, Wenbin Zhang, and Yanzhao Wu. 2023. Rethinking learning rate tuning in the era of large language models. In *2023 IEEE 5th International Conference on Cognitive Machine Intelligence (CogMI)*, pages 112–121. IEEE.
- Denis Kocetkov, Raymond Li, Loubna Ben Allal, Jia Li, Chenghao Mou, Carlos Muñoz Ferrandis, Yacine Jernite, Margaret Mitchell, Sean Hughes, Thomas Wolf, et al. 2023. The Stack: 3 TB of permissively licensed source code. *Transactions on Machine Learning Research*.
- Komal Kumar, Tajamul Ashraf, Omkar Thawakar, Rao Muhammad Anwer, Hisham Cholakkal, Mubarak Shah, Ming-Hsuan Yang, Phillip H.S. Torr, and Fahad Shahbaz Khan. 2025. LLM post-training: A deep dive into reasoning large language models. *arXiv preprint arXiv:2502.21321*.
- Hugo Laurençon, Lucile Saulnier, Thomas Wang, Christopher Akiki, Albert Villanova del Moral, Teven Le Scao, Leandro Von Werra, Chenghao Mou, et al. 2022. **The BigScience ROOTS corpus: A 1.6TB composite multilingual dataset**. *Advances in Neural Information Processing Systems*, 35:31809–31826.
- Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2022. Deduplicating training data makes language models better. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 8424–8445.
- Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Gadre, Hritik Bansal, Etash Guha, Sedrick Keh, et al. 2024a. DataComp-LM: In search of the next generation of training sets for language models. *arXiv preprint arXiv:2406.11794*.

- Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. 2024b. From quantity to quality: Boosting LLM performance with self-guided data selection for instruction tuning. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7602–7635.
- Hao Liang, Xiaochen Ma, Zhou Liu, Zhen Hao Wong, Zhengyang Zhao, Zimo Meng, Runming He, Chengyu Shen, Qifeng Cai, Zhaoyang Han, et al. 2025. DataFlow: An LLM-driven framework for unified data preparation and workflow automation in the era of data-centric AI. *arXiv preprint arXiv:2512.16676*.
- Chris Yuhao Liu, Liang Zeng, Yuzhen Xiao, Jujie He, Jiakai Liu, Chaojie Wang, Rui Yan, Wei Shen, Fuxiang Zhang, Jiacheng Xu, Yang Liu, and Yahui Zhou. 2025. Skywork-Reward-V2: Scaling preference data curation via human-AI synergy. *arXiv preprint arXiv:2507.01352*.
- Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. 2024. [What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning](#). In *International Conference on Learning Representations*.
- Sachin Mehta, Mohammad Hossein Sekhavat, Qingqing Cao, Maxwell Horton, Yanzi Jin, Chenfan Sun, Iman Mirzadeh, Mahyar Najibi, Dmitry Belenko, Peter Zatloukal, and Mohammad Rastegari. 2024. OpenELM: An efficient language model family with open training and inference framework. *arXiv preprint arXiv:2404.14619*.
- Meta AI. 2025. Llama 4: Multimodal intelligence. <https://ai.meta.com/blog/llama-4-multimodal-intelligence/>. Accessed: 2025-04-05.
- Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391.
- Kaixiang Mo, Yuxin Shi, Weiwei Weng, Zhiqiang Zhou, Shuman Liu, Haibo Zhang, and Anxiang Zeng. 2025. Mid-training of large language models: A survey. *arXiv preprint arXiv:2510.06826*.
- Team OLMo, Allyson Ettinger, Amanda Bertsch, Bailey Kuehl, David Graham, David Heineman, Dirk Groeneveld, Faeze Brahman, Finbarr Timbers, Hamish Ivison, et al. 2025. OLMo 3. *arXiv preprint arXiv:2512.13961*.
- Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2024. 2 OLMo 2 furious. *arXiv preprint arXiv:2501.00656*.
- OpenAI. 2024. Introducing improvements to the fine-tuning API and expanding our custom models program. <https://openai.com/index/introducing-improvements-to-the-fine-tuning-api-and-expanding-our-custom-models-program/>.
- OpenAI. 2025. OpenAI GPT-5 System Card. *arXiv preprint arXiv:2601.03267*.
- Guilherme Penedo, Hynek Kydlíček, Loubna Ben al-lal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. 2024. The FineWeb datasets: Decanting the web for the finest text data at scale. *arXiv preprint arXiv:2406.17557*.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-BERT: Sentence embeddings using Siamese BERT-networks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.
- Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. [WinoGrande: An adversarial Winograd schema challenge at scale](#). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8732–8740.
- Dan Su, Kezhi Kong, Ying Lin, Joseph Jennings, Brandon Norrick, Markus Kliegl, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. 2024. Nemotron-CC: Transforming common crawl into a refined long-horizon pretraining dataset. *arXiv preprint arXiv:2412.02595*.
- Yuan Sui, Mengyu Zhou, Mingjie Zhou, Shi Han, and Dongmei Zhang. 2024. [Table meets LLM: Can large language models understand structured table data? a benchmark and empirical study](#). In *Proceedings of the 17th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 645–654.
- Guiyao Tie, Zeli Zhao, Dingjie Song, Fuyang Wei, Rong Zhou, Yurou Dai, Wen Yin, Zhejian Yang, Jiangyue Yan, et al. 2025. A survey on post-training of large language models. *arXiv preprint arXiv:2503.06072*.
- Zhichao Wang, Kiran Ramnath, Bin Bi, Shiva Kumar Pentiyala, Sougata Chaudhuri, Shubham Mehrotra, Zixu James Zhu, Xiang-Bo Mao, et al. 2024. Reinforcement learning for LLM post-training: A survey. *arXiv preprint arXiv:2407.16216*.
- Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom

- Aarsen, Griffin Thomas Adams, Jeremy Howard, and Iacopo Poli. 2025. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2526–2547.
- Maurice Weber, Daniel Fu, Quentin Anthony, Yonatan Oren, Shane Adams, Anton Alexandrov, Xiaozhong Lyu, Huu Nguyen, Xiaozhe Yao, et al. 2024. Red-Pajama: an open dataset for training large language models. *arXiv preprint arXiv:2411.12372*.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Shuo Yang, Wei-Lin Chiang, Lianmin Zheng, Joseph E. Gonzalez, and Ion Stoica. 2023. Rethinking benchmark and contamination for language models with rephrased samples. *arXiv preprint arXiv:2311.04850*.
- Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.
- Han Zhang, Yuheng Ma, and Hanfang Yang. 2025a. [AL-TER: Augmentation for large-table-based reasoning](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 179–198, Albuquerque, New Mexico. Association for Computational Linguistics.
- Han Zhang, Langshi Zhou, and Hanfang Yang. 2025b. Learning to retrieve and reason on knowledge graph through active self-reflection. *arXiv preprint arXiv:2502.14932*.
- Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. 2024. TinyLlama: An open-source small language model. *arXiv preprint arXiv:2401.02385*.
- Xinlin Zhuang, Jiahui Peng, Ren Ma, Yinfan Wang, Tianyi Bai, Xingjian Wei, Qiu Jiantao, Chi Zhang, Ying Qian, and Conghui He. 2025. [Meta-rater: A multi-dimensional data selection method for pre-training language models](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10856–10896, Vienna, Austria. Association for Computational Linguistics.

A Data Crawling and Preprocessing

We crawl publicly accessible text datasets hosted on the Hugging Face Hub up to June 2025, excluding multi-modal datasets from the crawling scope. All collected data are stored in tabular form, where each dataset consists of multiple typed columns. We treat each split (e.g., train, validation, test) as the atomic unit of dataset operations: all splits are initially downloaded in Parquet format and subsequently converted to JSONL after the format standardization in Step 3. To keep downstream processing tractable, the size of each split is capped at 200 GB. To prevent test-data leakage, we discard all validation and test splits via regular-expression matching on split names. We further exclude organizations that contribute anomalously large numbers of Parquet files, as manual inspection reveals that such uploads predominantly consist of noisy cloud-storage backups that are intractable to process. After these preliminary filtering steps, we obtain a large and highly diverse initial pool spanning a wide range of domains, languages, and tasks.

B Implementation Details

Dataset-Level Processing Details. We provide details of the implementation and hyper-parameters used in Steps 2–6 in the dataset-level processing pipeline. For all sampling operations at this stage, we employ reservoir sampling to draw K rows uniformly at random from each dataset. The sampling budget is progressively increased as the number of remaining datasets decreases: following Zhang et al. (2025a), only $K=3$ samples are drawn in Steps 2 and 3, whereas $K=200$ is used in Steps 4 and 5, and $K=2000$ in Step 6. In Steps 2 and 3, we adopt the GPT-4.1 series as the backbone LLM for both coarse quality scoring and transformation function generation. In Step 4, the confidence threshold for language identification is set to 0.6, slightly more permissive than that of FineWeb (Penedo et al., 2024), as our goal is dataset-level language detection rather than strict per-sample filtering. In Step 5, we employ all-MiniLM-L6-v2 (Reimers and Gurevych, 2019) as the base embedding model, and the thresholds for the exact-duplicate ratio and the average semantic similarity are set to 0.30 and 0.95, respectively. In Step 6, the overall scoring framework follows OpenDataArena (Cai et al., 2025), where the *Relevance* and *Correctness* dimensions are evaluated by

Table 4: Data statistics at each pipeline stage. *Token count estimated using a bytes-per-token ratio calibrated on random sub-samples.

Operation	#Datasets	#Tokens
Raw Crawl	206,123	18.4 T*
4-Dim Filtering	95,958	8.4 T
Language Identification	60,409	6.5 T
Intra-Dataset Duplication Detection	46,014	4.9 T
Fine-grained Quality Scoring	19,200	1.6 T
HuggingBrew-DRAUGHT	—	1.6 T
HuggingBrew-CRAFT	—	450 B
HuggingBrew-RESERVE	—	40 B

the GPT-4.1 series as the LLM-as-Judge. We also provide the prompt templates used in the Step 2 and 3, which are correspondingly illustrated in Figure 13 and Figure 14.

Mid-Training Details. To ensure a fair comparison in the mid-training experiments, all training is performed in bfloat16 on 8 nodes $\times$ 8 NVIDIA A100-40GB GPUs connected via NVLink and InfiniBand, using Hugging Face Accelerate with DeepSpeed ZeRO Stage-1 and a fused AdamW optimizer. We use a per-device batch size of 8, yielding a global batch size of 512 sequences per optimization step. The learning rate is set to 2×10^{-4} and decayed by a cosine schedule with a 0.5% linear warmup. We further tune the cosine cycle length to prevent the learning rate from decaying to true zero.

Pipeline Statistics. Table 4 reports the number of retained datasets and tokens after each pipeline step, illustrating the progressive contraction from the initial 18.4 T-token raw crawl to a 1.6 T-token Tier-A corpus before global deduplication, and finally to the 450 B-token HUGGINGBREW-CRAFT and 40 B-token HUGGINGBREW-RESERVE corpora after MinHash deduplication. Figure 6 summarizes the resulting language distribution before Step 4, where English-dominant datasets account for the vast majority of the surviving collection, consistent with the overall composition of the Hugging Face ecosystem.

C Dataset Licensing and Compliance

We conduct a rigorous license audit over all candidate datasets following The Stack (Kocetkov et al., 2023): copyleft licenses such as GPL are conservatively excluded because their viral nature can impose unintended redistribution obligations on derivative works.

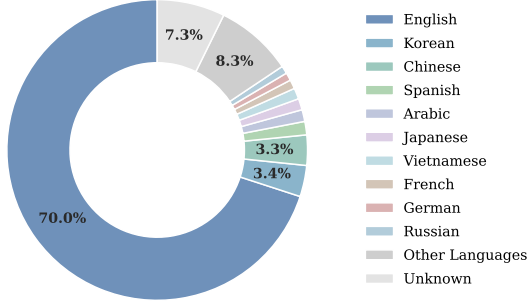


Figure 6: Distribution of languages across the curated dataset.

Filtering criteria. We filter datasets which explicitly prevents authorization for redistribution and derivative works. Four categories are filtered out: (i) **NC/ND clauses** (e.g., CC-BY-NC-4.0, CC-BY-ND-4.0); (ii) **strong copyleft** licenses (GPL, AGPL, GFDL) whose applicability to data is unsettled; (iii) **model-specific or use-restricted licenses** (Llama Community, Gemma Terms, OpenRAIL variants) imposing acceptable-use or competitive-use prohibitions; and (iv) **unresolvable licenses** that cannot be programmatically verified. Table 5 lists the resulting inventory of retained licenses.

D Fine-grained Quality Filtering Rules

We instantiate two rule sets over the Step-6 five-dimensional scores. Table 8 defines the **standard** rule set: six lightweight criteria that grade each dataset into Tiers A/B/C (used to assemble HUGGINGBREW-CRAFT). Table 9 defines the **stringent** rule set: 12 Tier-A admission rules that must *all* be satisfied (used for HUGGINGBREW-RESERVE). Together they realize the quality-scale trade-off discussed in §4.3.

E Format Standardization Examples

To illustrate how the format standardization handles heterogeneous schemas, we present four representative examples spanning code commits, question-answering, summarization, and biomedical retrieval datasets. Each example shows the original dataset schema alongside the canonicalized output with three fields: premise, brief, and artifact. As shown in Figures 9 to 12, the transformation functions adapt to diverse column structures and semantic conventions while producing a unified schema suitable for downstream processing.

Table 5: Inventory of permissible licenses retained in the curated corpus, grouped by category.

License (SPDX)	Description
<i>Permissive</i>	
Apache-2.0	Patent grant; attribution required
MIT	Minimal restrictions; attribution only
CC-BY-4.0/3.0/2.5/2.0	Attribution; free to adapt & share
CC0-1.0	Public domain dedication
ODC-BY	Open data; attribution required
AFL-3.0	Academic free license; patent grant
BSD-3/2-Clause	Attribution; no endorsement clause
Unlicense	Public domain equivalent
WTFPL	No restrictions whatsoever
PDDL	Public domain for data
CDLA-Permissive-2.0/1.0	Data-specific; attribution only
Etalab-2.0	French government open data license
BSL-1.0	Boost Software License; permissive
ISC / PostgreSQL	Simplified BSD-style
MS-PL	Microsoft permissive license
ECL-2.0	Educational community license
<i>Share-Alike</i>	
CC-BY-SA-4.0/3.0	Attribution; share-alike required
ODbL-1.0	Open database; share-alike
CDLA-Sharing-1.0	Data-specific; share-alike required
<i>Weak Copyleft</i>	
MPL-2.0	File-level copyleft; permissive linking
EURL-1.1	EU public license; weak copyleft
EPL-2.0	Eclipse public license
OSL-3.0	Open Software License
Artistic-2.0	Modified distribution permitted
<i>Under-specified</i>	
CC	Version not declared; treated as CC-BY
BSD	Clause not declared; treated as BSD-3

F Ablation Study on Scoring Model

We further validate our scorer design with two ablations: encoder input length and single- vs. multi-head architecture. The distillation model uses ModernBERT-large (up to 8192 tokens) as a shared encoder; the [CLS] embedding is projected through a shared MLP and fed into five independent linear heads predicting Quality, Reward, IFD, Correctness, and Relevance. Training uses 100K samples drawn evenly from the Hugging Face pool, with ground-truth labels from model-based scorers and GPT-4.1 LLM-as-Judge evaluators. IFD scores are first hard-capped at 100 and $\log(1+x)$ -transformed; all dimensions are then z -score normalized during training and inverse-transformed back to their original scale at inference time.

Ablation on Encoder Input Length. We conduct ablation experiments with three input length configurations—2048, 4096, and 8192 tokens—while keeping all other hyperparameters fixed. Table 6 reports the Pearson and Spearman correlation coefficients between predicted and ground-truth scores for each input length configuration. Across all five scoring dimensions, the 2048-token configu-

Table 6: Correlation between the distilled scorer’s predictions and the ground-truth scores at three encoder context lengths.

Dimension	Pearson $r \uparrow$			Spearman $\rho \uparrow$		
	2048	4096	8192	2048	4096	8192
IFD	.911	.664	.428	.891	.601	.481
Quality	.956	.938	.881	.953	.929	.872
Reward	.843	.724	.594	.839	.718	.588
Correctness	.771	.607	.441	.749	.597	.422
Relevance	.777	.560	.442	.734	.550	.418
Average	.852	.698	.557	.833	.679	.556

Table 7: MSE of the multi-head model versus separate models distillation architectures on the in-distribution (ID) and out-of-distribution (OOD) test sets.

Dimension	ID (MSE $\downarrow$)		OOD (MSE $\downarrow$)	
	Multi	Single	Multi	Single
IFD	.0034 $+.0004$	.0030	.3549 $+.0335$	.3214
Quality	.0054 $+.0018$	.0036	.0068 $+.0020$	.0048
Reward	.0116 $+.0007$	.0109	.0169 $+.0008$	.0161
Correctness	.0447 $-.0024$	.0471	.0568 $-.0047$	.0615
Relevance	.0485 $-.0021$	.0506	.0635 $-.0041$	.0675
Average	.0227 $-.0003$	.0231	.0998 $+.0055$	.0943

ration consistently achieves the highest correlation, outperforming the 4096 and 8192 settings by a substantial margin. These results suggest that shorter input lengths lead to more effective score prediction in our setting.

Multi-Head vs. Single-Head Architecture. We further compare two architectural choices for the distillation model: (1) a *multi-head* architecture that shares a single encoder backbone and uses a shared MLP layer followed by five independent linear heads, and (2) a *single-head* baseline that trains five entirely separate single-output models, one per scoring dimension. We report the mean squared error (MSE) on both in-distribution (ID) and out-of-distribution (OOD) test sets in Table 7, where the OOD set consists of 5,000 samples drawn from a separate held-out pool that was not used for training. On average, the two architectures perform comparably on both ID and OOD data, while the multi-head model offers substantially greater inference efficiency.

Four-Dimensional Quality Assessment. We conduct a statistical analysis of the four-dimensional scores assigned during Step 2 filtering. As shown in Figure 7, the predefined thresholds

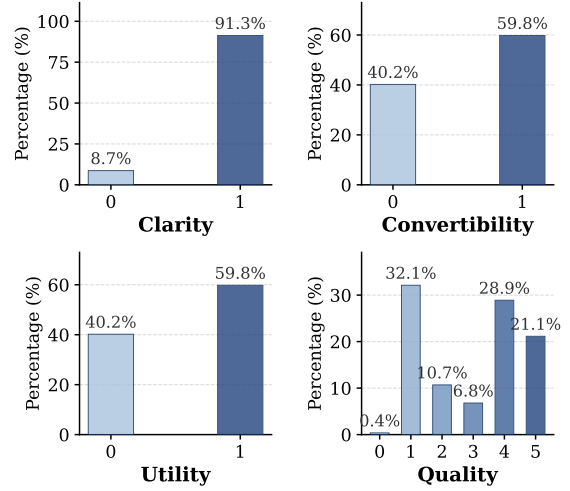


Figure 7: Distribution of scores across the four-dimensional filtering criteria.

effectively remove clearly low-quality datasets—such as those with a clarity score of 0 or a quality score ≤ 1 —resulting in the retention of approximately half of the original datasets.

G Domain Composition

Drawing on the subject taxonomy of MMLU (Hendrycks et al., 2021a) and Wikipedia’s knowledge categorization, we characterize the final corpus using eight mutually exclusive high-level domains. To support scalable annotation, we fine-tune a DeBERTa-v3 domain classifier (He et al., 2023) on 20,000 examples labeled by GPT-4.1. As shown in Figure 8, Mathematics and Technology & Engineering account for the largest shares, followed by Humanities & Social Sciences and Natural Science, while the remaining domains provide broad coverage of arts, business, health, law, and ethics.

H Decontamination List

To prevent test-set leakage into the training corpus, we perform n -gram-based ($n = 12$) decontamination (Step 9 in §3.3) against *all* evaluation subsets used in our work, following the protocol of DCLM (Li et al., 2024a). Table 10 enumerates the full set of 72 benchmark subsets, organized by parent benchmark. The list spans general knowledge (MMLU, HellaSwag, OpenBookQA, WinoGrande, ARC, PIQA), mathematical reasoning (GSM8K, MATH), and code generation (HumanEval), with MMLU contributing 57 subjects and MATH contributing 7 topical splits.

Table 8: Dataset-level grading rules for the standard rule set used by HUGGINGBREW-CRAFT. All rules are expressed as *satisfaction* conditions; a dataset is classified into Tier A (all rules satisfied), Tier B (exactly one rule violated), or Tier C (≥ 2 rules violated).

#	Rule	Metric	Threshold	Rationale
1	Quality Diversity & Mean	$\max_b \frac{ \{x: q(x) =b\} }{N} \& \bar{q}$	$< 0.90 \vee \bar{q} > 3$	Avoid template-like low-quality data
2	Reward Stability & Mean	$CV_{\text{reward}} \& \bar{r}$	$CV \leq 0.8 \vee \bar{r} \geq 8.0$	Avoid noisy and low-reward distributions
3	Avg. Output Tokens	$\bar{l}_{\text{tok}}$	≥ 30	Ensure sufficient information density
4	IFD Mean	$\overline{\text{IFD}}$	≥ 0.3	Ensure non-trivial instructions
5	Correctness Mean	$\overline{\text{Corr}}$	≥ 2.0	Ensure factual accuracy
6	Relevance Mean	$\overline{\text{Rel}}$	≥ 2.0	Ensure instruction relevance

Table 9: Dataset-level grading rules for the stringent rule set used by HUGGINGBREW-RESERVE. A dataset is admitted to Tier A only if *all* of the following rules are satisfied.

#	Rule	Metric	Threshold	Rationale
1	Quality Mean	$\bar{q}$	≥ 3.50	Ensure high average quality score
2	Quality Median	$\tilde{q}$	≥ 3.30	Ensure robust quality across samples
3	Quality Diversity & Mean	$\max_b \frac{ \{x: q(x) =b\} }{N} \& \bar{q}$	$< 0.7 \vee \bar{q} \geq 4.0$	Allow concentration only for very high-quality data
4	Reward Mean	$\bar{r}$	≥ 16.0	Ensure high overall reward
5	Reward Q25	r_{q25}	≥ 12.0	Ensure strong reward lower tail
6	Reward Stability & Mean	$CV_{\text{reward}} \& \bar{r}$	$CV \leq 0.45 \vee \bar{r} \geq 20.0$	Allow noisier reward only when mean is very high
7	IFD Mean	$\overline{\text{IFD}}$	≥ 0.65	Ensure non-trivial instructions
8	IFD Median	$\overline{\text{IFD}}$	≥ 0.60	Ensure robust instruction difficulty
9	Correctness Mean	$\overline{\text{Corr}}$	≥ 8.0	Ensure high factual accuracy
10	Relevance Mean	$\overline{\text{Rel}}$	≥ 8.5	Ensure strong instruction alignment
11	Avg. Output Tokens	$\bar{l}_{\text{tok}}$	≥ 60	Ensure sufficient output informativeness
12	Median Output Tokens	$\tilde{l}_{\text{tok}}$	≥ 60	Ensure consistent output length

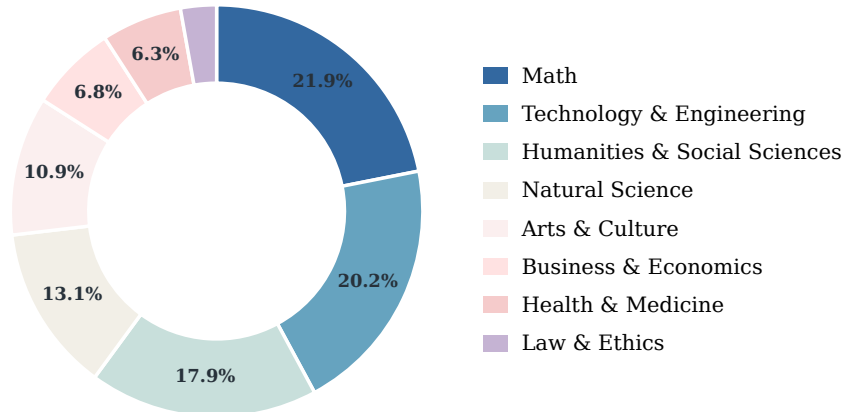


Figure 8: Document-level domain distribution of HUGGINGBREW-RESERVE. Each document is assigned to exactly one of the eight high-level domains.

Table 10: Decontamination targets covering all 72 evaluation subsets used in our experiments.

Benchmark	Count	Evaluation subsets
ARC (Clark et al., 2018)	2	ARC-Challenge, ARC-Easy
GSM8K (Cobbe et al., 2021)	1	GSM8K
HellaSwag (Zellers et al., 2019)	1	HellaSwag
HumanEval (Chen et al., 2021)	1	HumanEval
MATH (Hendrycks et al., 2021b)	7	Algebra, Counting & Probability, Geometry, Intermediate Algebra, Number Theory, Prealgebra, Precalculus
MMLU (Hendrycks et al., 2021a)	57	All 57 official subject categories, from <i>abstract algebra</i> to <i>world religions</i>
OpenBookQA (Mihaylov et al., 2018)	1	OpenBookQA
PIQA (Bisk et al., 2020)	1	PIQA
WinoGrande (Sakaguchi et al., 2020)	1	WinoGrande
Total	72	

Example 1: Code Commit Dataset

• Original Schema

```
{
  "commit": "e905334...",
  "old_file": "sentry/queue/client.py",
  "new_file": "sentry/queue/client.py",
  "old_contents": "class Broker(object):\n  def __init__(self, config): ...",
  "new_contents": "class Broker(object):\n  def __init__(self, config): ...
with producers[...].acquire(...) as producer:
for queue in task_queues: maybe_declare(...) ...",
  "subject": "Declare queues when broker is instantiated",
  "message": "Declare queues when broker is instantiated",
  "lang": "Python", "license": "bsd-3-clause", "repos": "imankulov/sentry, ..."
}
```

• Canonicalized Output

```
premise: File: sentry/queue/client.py
brief: Before:
class Broker(object):
def __init__(self, config):
self.connection = BrokerConnection(**config)
...
After:
class Broker(object):
def __init__(self, config):
self.connection = BrokerConnection(**config)
with producers[self.connection].acquire(block=False) as producer:
for queue in task_queues: maybe_declare(queue, producer.channel)
...
artifact: Declare queues when broker is instantiated
```

Figure 9: Format standardization example for a **code commit** dataset.

Example 2: Question-Answering Dataset

• Original Schema

```
{
  "Unnamed: 0": 0,
  "Question": "What is under-fitting and overfitting in machine learning?",
  "Answer": "Underfitting is when a model is too simple, and overfitting is when it's too complex, making it perform poorly on new data."
}
```

• Canonicalized Output

premise: (empty)

brief: What is under-fitting and overfitting in machine learning?

artifact: Underfitting is when a model is too simple, and overfitting is when it's too complex, making it perform poorly on new data.

Figure 10: Format standardization example for a **question-answering** dataset.

Example 3: Summarization Dataset

• Original Schema

```
{
  "articles": "Faith schools citizenship warning. Schools must improve the quality of citizenship lessons – or social cohesion and democracy will suffer, says the education watchdog. Independent faith schools were singled out by Ofsted chief, David Bell, for not doing enough to promote the \"wider tenets of British society\". Mr Bell said Muslim, Jewish and Evangelical Christian schools must be \"intolerant of intolerance\". Diversity \"certainly must not mean segregated or separate\", he said...",
  "summaries": "Schools must improve the quality of citizenship lessons or social cohesion and democracy will suffer, says Ofsted chief David Bell."
}
```

• Canonicalized Output

premise: (empty)

brief: Faith schools citizenship warning. Schools must improve the quality of citizenship lessons – or social cohesion and democracy will suffer, says the education watchdog. Independent faith schools were singled out by Ofsted chief, David Bell, for not doing enough to promote the “wider tenets of British society”. ...

artifact: Schools must improve the quality of citizenship lessons or social cohesion and democracy will suffer, says Ofsted chief David Bell.

Figure 11: Format standardization example for a **summarization** dataset. The articles column is mapped to brief and the summaries column to artifact.

Example 4: Biomedical Retrieval Dataset

• Original Schema

```
{
  "query": ["prolonged QT interval, congenital deafness",
    "abnormal cardiac ventricular repolarization, bilateral congenital deafness",
    "QT interval prolongation, hearing loss",
    "cardiac repolarization disorder, congenital deafness, KCNE1 gene mutation"],
  "chunk": "A number sign (#) is used with this entry because of evidence that
    Jervell and Lange-Nielsen syndrome-2 (JLNS2) is caused by homozygous or
    compound heterozygous mutation in the KCNE1 gene (176261) on chromosome
    21q22. ... The Jervell and Lange-Nielsen syndrome is an autosomal recessive
    syndrome of abnormal cardiac ventricular repolarization with prolonged QT
    interval and bilateral congenital deafness. ..."
}
```

• Canonicalized Output

premise: (empty)

brief: prolonged QT interval, congenital deafness, ...

artifact: A number sign (#) is used with this entry because of evidence that JLNS2 is caused by homozygous or compound heterozygous mutation in the KCNE1 gene (176261) on chromosome 21q22. ... The Jervell and Lange-Nielsen syndrome is an autosomal recessive syndrome of abnormal cardiac ventricular repolarization with prolonged QT interval and bilateral congenital deafness. ...

Figure 12: Format standardization example for a **biomedical retrieval** dataset.

Dataset-Level Quality Evaluation Prompt

You are an expert data curator tasked with filtering and selecting high-quality mid-training data for a large language model. Your goal is to identify datasets that enhance the model's ability to perform tasks such as knowledge grounding, instruction-following, complex reasoning, extraction, generation, etc. The selected datasets should be transformable into standardized input queries and corresponding output responses for model training.

Evaluate each dataset (not individual samples) based on the following criteria:

1. **Clarity** (0/1): Assess whether the dataset's content is logically clear, unambiguous, and intuitively understandable. The transformed data should convey its purpose and context with explicit semantic meaning, requiring minimal interpretation. Assign a score of 1 if the dataset meets this criterion, or 0 if it does not.
2. **Convertibility** (0/1): Verify that the dataset's schema can be programmatically mapped onto canonical three-field format (premise, brief, artifact). Exclude purely non-textual data (e.g., embedding arrays) and irreducibly nested structures whose semantics cannot be flattened into coherent text. When multiple columns are present, use your best judgment to align them with the target fields. Assign a score of 1 if the dataset can be reliably transformed, or 0 otherwise.
3. **Utility** (0/1): Prioritize datasets with high educational value that enhance the model's capabilities. The content should provide meaningful learning opportunities through fluent text, such as improving commonsense reasoning, factual knowledge, reading comprehension, mathematical reasoning, problem-solving, long-context understanding, or code generation. Assign a score of 1 if the dataset offers significant training value, or 0 if it does not.
4. **Quality** (0–5): Conduct a comprehensive evaluation of the dataset's overall quality, including but not limited to diversity, utility, completeness, complexity, etc. Provide a comprehensive quality score from 0 to 5, where a higher score indicates superior overall quality.

Provide a concise (2–4 sentence) summary of your evaluation, describing the dataset's structure, content type, and potential applications for large language model training or evaluation. Ensure your evaluation is clear, actionable, and optimized for building a robust, high-quality training corpus.

You will be given sampled rows in TOML format, consisting of a set of key-value fields (some flat, some nested).

Samples: `{samples}`;

Output your final evaluation for the dataset.

Figure 13: Prompt template for dataset-level quality evaluation. We additionally provide in-context examples (omitted for space) to calibrate scoring standards.

Format Standardization Prompt

You are an expert Data Engineer specializing in NLP dataset preparation for Large Language Models (LLMs). Your task is to analyze the structure of a given dataset and write a Python code snippet to transform a single row of data into a standardized dictionary format.

Dataset Information:

Summary: *{summary}*;

Columns: *{columns}*;

Column schemas: *{schemas}*;

Sample data (3 rows): *{samples}*.

Target Format. The output dictionary result must have exactly these keys:

1. **premise** (str): Background information, context, or preface. Defaults to empty string if not present.
2. **brief** (str): The primary content or query. If multiple fields contribute (e.g., title and content), you can combine them logically.
3. **artifact** (str): The expected follow-up, response, answer, or label.
4. **append** (dict): A dictionary containing all other metadata fields not part of the text training data but worth preserving.

Transformation Guidelines:

1. **Semantic Analysis:** Infer column meaning from the dataset information
2. **Special Cases:** (a) Single-column: route or split the text across target fields by semantic role; leave unused fields empty. (b) Context concatenation: combine context and query with clear separators. (c) Conversations: extract system message if present; use prior history as brief and last assistant turn as artifact. (d) Preference data: set artifact to the best available response.
3. **Robustness:** Handle missing keys via `.get()`; convert None to empty string; ensure all values in premise, brief, artifact are strings.
4. **Cleanup:** Exclude technical fields (e.g., id, index, uuid) unless semantically informative.

Output Requirement: Provide only the Python code block operating on a variable row (a dictionary) and populating the existing dictionary result. No function definitions or explanations outside the code.

Figure 14: Prompt template for model-driven format standardization. The LLM receives dataset metadata and sample rows, then generates a Python transformation function that maps heterogeneous feature columns into the canonical four-field schema (premise, brief, artifact, append). The append field preserves meaningless and confusing schema, excluded from downstream corpus processing and model training.