Unleashing the Power of Data Tsunami: A Comprehensive Survey on Data Assessment and Selection for Instruction Tuning of Language Models

Anonymous authors
Paper under double-blind review

Abstract

Instruction tuning plays a critical role in aligning large language models (LLMs) with human preference. Despite the vast amount of open instruction datasets, naively training a LLM on all existing instructions may not be optimal and practical. To pinpoint the most beneficial datapoints, data assessment and selection methods have been proposed in the fields of natural language processing (NLP) and deep learning. However, under the context of instruction tuning, there still exists a gap in knowledge on what kind of data evaluation metrics can be employed and how they can be integrated into the selection mechanism. To bridge this gap, we present a comprehensive review on existing literature of data assessment and selection especially for instruction tuning of LLMs. We systematically categorize all applicable methods into quality-based, diversity-based, and importance-based ones where a unified, fine-grained taxonomy is structured. For each category, representative methods are elaborated to describe the landscape of relevant research. In addition, comparison between the latest methods is conducted on their officially reported results to provide in-depth discussions on their limitations. Finally, we summarize the open challenges and propose the promosing avenues for future studies. All related contents are available at `https://anonymous.4open.science/status/instruction_tuning_data_survey-B307`.

1 Introduction

One of the ultimate goal of developing large lnguage models (LLMs) is to unlock their potentials of generalization to unseen natural language processing (NLP) tasks. Towards this goal, a series of LLMs such as GPTs [Brown et al. (2020); Achiam et al. (2023)], LLaMAs [Touvron et al. (2023a;b); AI@Meta (2024)], and Mistrals [Jiang et al. (2023a; 2024a)] have delivered high-level text understanding and generation capabilities via utilizing vast amount of high-quality web and human-annotated datasets for pre-training and preference alignment [Liu et al. (2023a; 2024c); Sun et al. (2024b); Edunov et al. (2019); Dong et al. (2019)]. During preference alignment, instruction tuning plays an important role in refining the pre-trained LLMs to provide accurate, pertinent, and harmless responses on a collection of downstream tasks [Wei et al. (2021); Sanh et al. (2021); Zhang et al. (2023d); Peng et al. (2023); Longpre et al. (2023); Shu et al. (2023); Jang et al. (2023); Ghosh et al. (2024); Kung & Peng (2023)]. For efficient and effective instruction tuning, existing studies [Ouyang et al. (2022); Taori et al. (2023); Zhou et al. (2024a); Xia et al. (2024a)] have noticed that improving the quality of instruction tuning data (e.g., formulation of well-defined and complete contexts), rather than simply piling up instructions without analysis (e.g., exhaustive collection of open datasets), is of prioritized concerns.

In this work, we aim to unify a wide array of data assessment and selection methods under the context of instruction tuning of LLMs. As revealed from the probabilistic view [John & Draper (1975); Murphy (2012); Albalak et al. (2024)], the statistical patterns inherent in datasets determine the modeling performance. The overall evaluation of datapoints not only deciphers the distribution in various aspects (e.g., composition, task, and domain) but also helps cherry-pick the most beneficial subsets for higher performance with less training cost. Through this survey, we demonstrate that: 1) existing resourceful data assessment methods can be

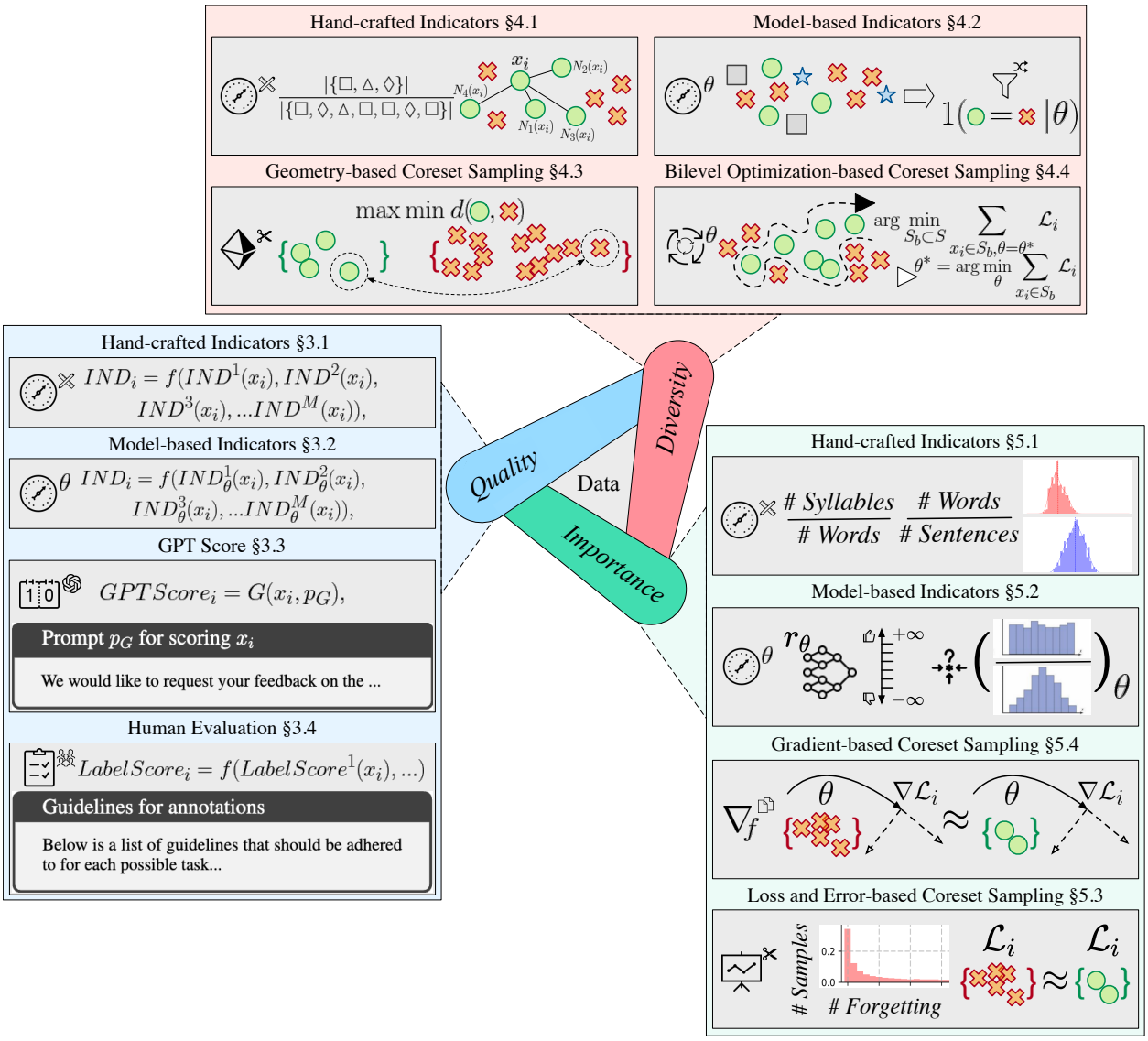

Figure 1: Categorization of data assessment and selection methods for effective instruction tuning of LLMs.

categorized into three main perspectives: quality, diversity, and importance (see Fig. 1). 2) a systematic view of selection methods can be unified even they more or less exhibit coupling with the assessment techniques (see Fig. 2). It is noted that quality, diversity, and importance might be used interchangeably without strict discrimination in previous studies. But here we provide a rationalized organization taxonomy for structured elaboration. Despite the goal of being comprehensive, the present survey only provides details of certain typical, representative methods to avoid being tediously long. We hope the in-depth explanations and discussions on the selected methods provide insights into developing robust data assessment and selection pipelines for future studies.

1.1 Related Surveys

[Liu et al. (2024d)] studied the mainstream datasets for building LLMs, including the pre-training corpora, instruction tuning datasets, preference datasets, evaluation benchmarks, and traditional NLP datasets. Their work focuses on the descriptions of dataset statistics (e.g., categorization, sources, and domains) without

providing guidelines on utilization. In contrast, we emphasize the selection of instruction-tuning data for the improved downstream performance. [Albalak et al. (2024)] presented a systematic overview of constructing the data pipeline for language models. Any selection method, either via distribution matching or diversification, can be composed of: 1) utility function; 2) selection mechanism. During different stages of the pipeline, the selection method should be adjusted according to different selection objectives (e.g., language filtering, data quality control , domain knowledge division , deduplication, toxic and explicit content removal, and data mixing). Their work pays extra attention to the processing of the pre-training corpora while neglecting the fine-grained analysis of existing selection methods specifically designed for instruction tuning. In this case, our survey serves as an indispensable extension on the selection of instruction datasets. [Wang et al. (2024a)] focused on the data preparation for instruction tuning. Existing methods on building instruction tuning datasets include: 1) reformulating the discriminative NLP datasets into generative ones; 2) self-instruct with seed prompts; 3) prompt mapping and evol-instruct Popular methods on dataset selection can be simply classified as: 1) system of indicators; 2) trainable LLMs; 3) powerful LLMs; and 4) small models. Comparatively, our survey stems from the characteristics of data themselves, namely quality, diversity, and importance, for categorization of selection methods. In each category, we further provide subdivided groups by the selection philosophy, which deepens the understanding of selection for practical take-home messages.

Existing surveys on general data selection also shed light on the principles of developing selection methods. [Guo et al. (2022)] started from the general coreset selection methods in the field of deep learning and categorize all selection manners into: 1) geometry-based methods (e.g., herding, k-center greedy); 2) uncertainty-based methods (e.g., least confidence/entropy/margin); 3) error/loss-based methods (forgetting; GraND/EL2N; importance resampling); 4) decision boundary-based (adversarial deepfool; contrastive active learning); 5) gradient matching-based (gradient approximation towards full set); 6) bi-level optimization-based (inner loop of model optimization and outer loop of datapoint selection); 7) sub-modularity-based (e.g., graph cut; facility location); 8) proxy-based (preference of a small model on data selection). [Zhou et al. (2024b)] investigated the potential metrics and aspects for data quality measurement. They provide a list of available tools for data evaluation. Apart from data assessment and selection methods that are specifically designed for NLP or LLM applications [Moore & Lewis (2010); Chen et al. (2024a); Dodge et al. (2020); Kandpal et al. (2022); Li et al. (2022); Feng et al. (2021); Lee et al. (2021); Malhotra & Bakal (2015); Liu et al. (2024e)], there exist many survey studies that tackle general quality measurement in machine learning [Gupta et al. (2021); Zha et al. (2023); Ehrlinger & Wöß (2022); Mohammed et al. (2024); Li et al. (2024c); Lu et al. (2023b); Dix et al. (2023); Priestley et al. (2023); Byabazaire et al. (2020); Roh et al. (2019); Sidi et al. (2012); Batini et al. (2009)] for constructing safe, unbiased, and accurate datasets. In this paper, we mainly focus on reviewing data selection methods for utilizing instruction tuning data in the context of large language models.

1.2 Survey Scope

Although "data evaluation" has been so frequently mentioned that it appears as a cliché problem in developing machine learning algorithms, the optimal solution to establishing an overall data assessment and selection pipeline still remains an open question. Especially under the context of instruction tuning of LLMs, existing studies proposed various measurements and strategies to select the "high-quality" instructions. However, very few studies noticed that there exists no unified dimensions or aspects in measuring data "quality" where previous works tend to put emphasis on the domain-specific and task-dependent characteristics. In addition, the inherent, systematic coupling between data assessment and subset selection is not well demonstrated.

Under such circumstance, the present study strives to provide a comprehensive review on evaluating and decomposing massive instruction tuning datasets. We categorize the main aspects of data assessment in terms of **quality**, **diversity**, and **importance**. To reduce ambiguity, their definitions are first provided below.

Quality refers to the intrinsic value of the data. High-quality data typically satisfy two conditions: 1) The instructions are clear, accurate, and explicit in explaining the task at hand and the expected behavior of LLMs. 2) The responses are correct, coherent, and pertinent to the instructions. All the requirements and constraints specified in instructions should be met. Accordingly, most existing methods not only underline the clarity and fluency of instructions but also filter out mismatched, incorrect, and harmful responses via various judging and measuring techniques.

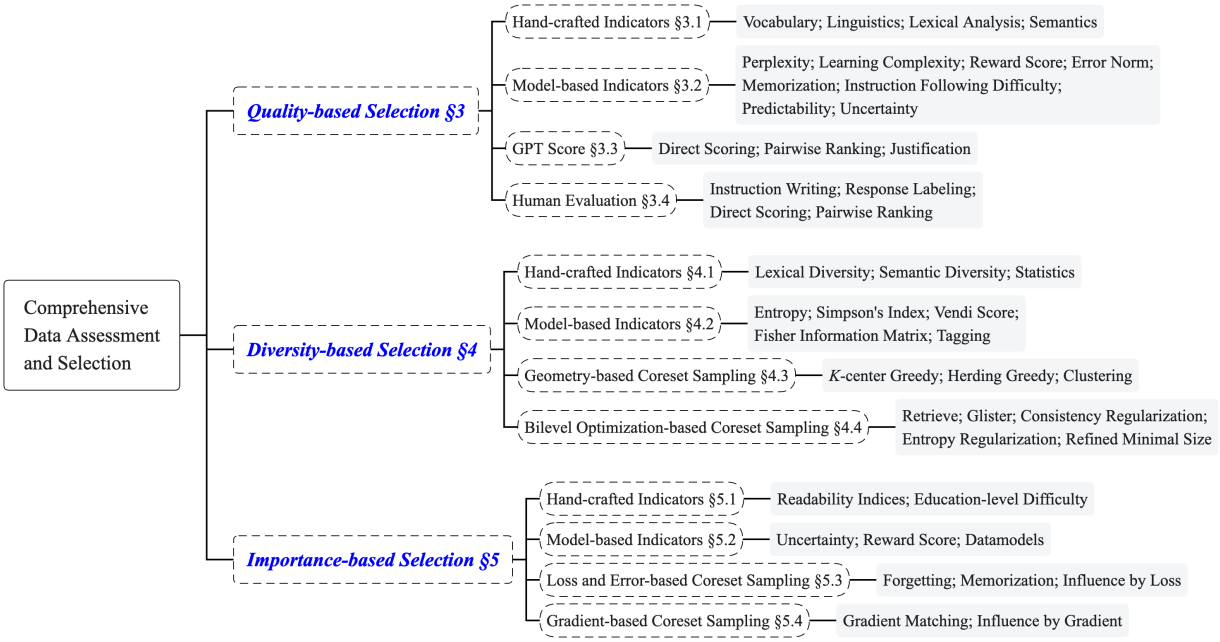

Figure 2: A high-level overview of comprehensive data assessment and selection. The analysis aspects that apply to either individuals or the overall dataset can be categorized into three groups marked in blue italic.

Diversity refers to the variety and richness of the dataset. During training, models that are exposed to datapoints under a wide range of scenarios enjoy a high level of generalization to unseen tasks. High-diversity data imply that each constituting datapoint is unique. The diverse dataset should cover different knowledge domains and task definitions with varying input contexts and output constraints. In light of this statement, existing methods that prioritize diversity all emphasize the removal of homogeneous, near-deduplicated datapoints in terms of lexical and semantic representations.

Importance refers to the impact of specific data points on the LLM's performance. It implies the necessity of adding one datapoint into the selected subset for instruction tuning. The importance of a datapoint is up to factors such as the difficulty of following the instruction, the capability of the LLM under investigation, and the scale of the dataset to be selected. Under such circumstance, various methods of importance-based selection have been developed by measuring the difficulty of each datapoint, matching the gradients or performance of models that are trained on the selected and the full set, and investigating the dynamics of models in learning rich and scarce datasets.

In each aspect, we provide a detailed survey on both traditional (e.g., hand-crafted indicators) and machine learning (e.g., model-based indicators) methods. Besides, the coreset sampling methods that bridge evaluation and selection are introduced separately in diversity and importance oriented subset construction. In consideration of the properties of instruction tuning, we focus on the text modality and start from classical text analysis metrics. Metrics that are either specific to instruction tuning or compatible with pre-training and preference alignment are included since they all share general rules in data assessment.

The survey is organized as follows. First, we present the preliminaries for assessment and selection of instruction tuning datasets (Sec. §2). Next, we present the surveying methods of data assessment and selection methods in terms of quality (Sec. §3), diversity (Sec. §4), and importance (Sec. §5). Then, discussions on the existing methods are provided in (Sec. §6), followed by the promising directions for future research (Sec. §7). The final conclusion is given in (Sec. §8).

1.3 What's Beyond the Survey Scope

The instruction tuning methods for de-bias and fairness of LLMs are not covered in the present study. We acknowledge that the bias and fairness aspects of instruction data are valued in developing responsible LLMs [Gallegos et al. (2024); Chu et al. (2024); Li et al. (2023d)]. However, they are beyond the scope of the present study for the following reasons:

- Most **data selection** methods on instruction tuning do not even notice the bias or fairness of data. They are in lack of explicitly designing steps to reduce negative impacts of biased data.
- Existing fairness tuning methods follow their own definitions of quality and diversity in terms of gender, race, religion, profession, age, and political ideology. It is difficult to bring in all their corresponding evaluation techniques under our selection taxonomy.
- The bias and fairness not only intersect with diversity but also with quality control, which is not an inclusive concept. For example, to control the harmful contents like hatred and racism responses, reward models or GPT scoring from quality-based selection can be used because the publicly released reward models and LLMs are already aligned with human preference.
- The evaluation of bias and fairness is not covered in existing studies on data selection, which makes it difficult to validate the effectiveness of selection techniques in improving the fairness of LLMs.

However, we emphasize that bias and fairness should be specifically handled because instruction-tuned LLMs tend to exhibit more bias than the pre-trained LLMs [Itzhak et al. (2024)]. Therefore, we believe the selection for a less-biased dataset would be a promising future direction in developing comprehensive selection methods.

2 Preliminaries

In this section, we briefly introduce the instruction tuning of LLMs and the problem statement for dataset assessment and selection.

Instruction Dataset Preparation In instruction tuning, each text sample I_i is usually composed of three parts: 1) instruction (either with or without system prompt), 2) input (can be empty), and 3) response.

For an off-the-shelf pre-trained LLM parameterized as θ, a pre-determined instruction template is used to wrap the I_i into a textual prompt p_i with special tokens like "`<|im_start|>`" and "`<|im_end|>`" for separation of roles (e.g., `system`, `user`, `assistant`, `function`, and `observation`) and their contents. It is noted that the prompt template for organizing a text sample I_i differs across model families. Especially, the special tokens are unique to the tokenizers and play an important role in differentiating multi-turn conversations and stopping model generations.Then, a LLM-associated tokenizer performs tokenization on the textual prompt p_i for a sequence of `token-id` integers: $x_i = [x_{i(1)}, x_{i(2)}, ..., x_{i(n)}]$, where $x_{i(j)}$ denotes the j-th token of x_i and n is the total number of tokens. Out of simplicity, the token sequence x_i can be simply split into two parts by the index t: 1) the instruction part ($x_{i(<t)}$) that is fed into the LLM without being involved in loss computation, and 2) the ground-truth response part ($x_{i(\geq t)}$) that is expected as the LLM output via minimizing its language modeling loss. Without losing generality, we use the terminology of "datapoints" throughout the paper to refer to the tokenized samples x_i.

The process of processing an instruction sample x_i is illustrated in Fig. 3. We choose one example from the Alpaca dataset [Taori et al. (2023)] and perform Qwen tokenization [Yang et al. (2024)] for demonstration.

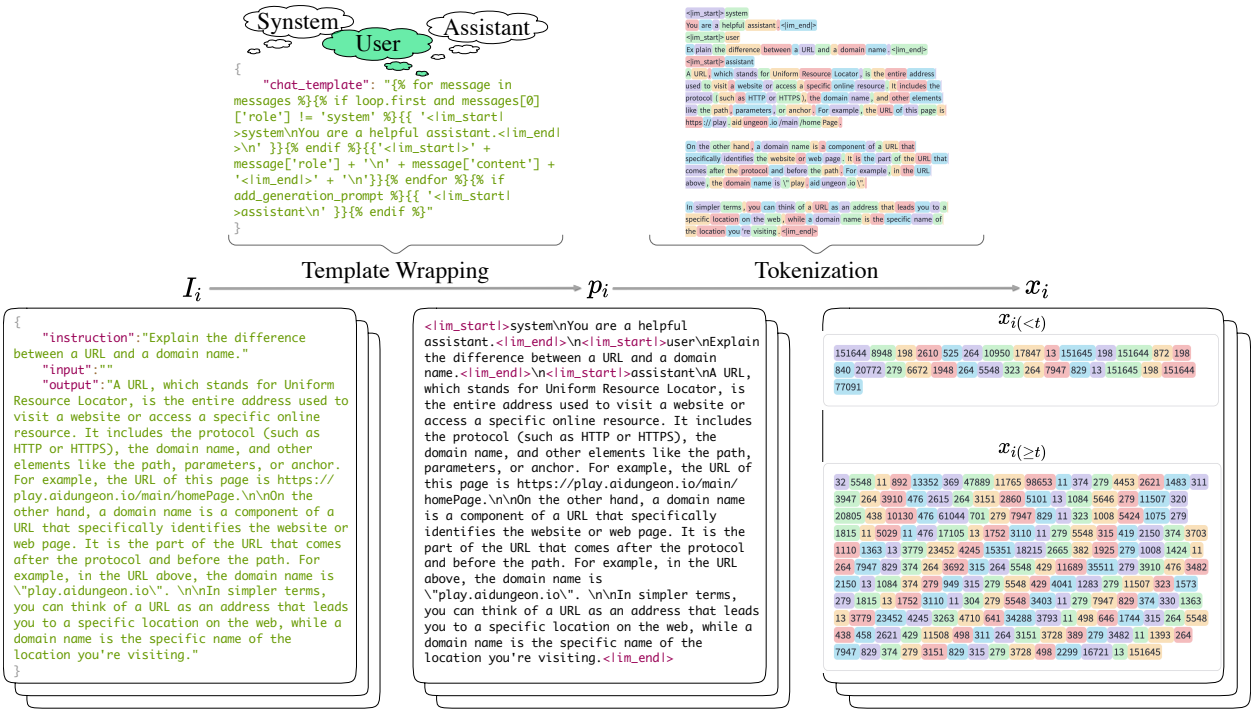

Figure 3: The pre-processing of an instruction dataset includes: 1) template wrapping, and 2) tokenization. In the first step, we wrap the raw texts I_i with a pre-defined chat template into the textual prompts p_i. In the second step, we perform tokenization on p_i with the LLM-associated tokenizer for the datapoint x_i. Given the index t for indicating where the loss mask of language modeling starts taking effect, we split x_i into $x_{i(<t)}$ and $x_{i(\geq t)}$, respectively denoting the instruction part (input) and the response part (output).

Instruction Supervision Given the tokenized instruction tuning dataset $S = \{x_i\}_{i=1}^{N}$, the supervised tuning is performed via cross-entropy loss:

$$\begin{aligned} \mathcal{L} &= \sum_{x_i \in S} \mathcal{L}_i, \\ \mathcal{L}_i &= -\sum_{j=t}^{|x_i|} \log P(x_{i(j)} | x_{i(<j)}; \theta). \end{aligned} \tag{1}$$

For each x_i, given all previous tokens $x_{i(<j)}$, the model iteratively predicts the next token $x_{i(j)}$ at the j-th index. Note that the previous tokens here include both the instruction context part $x_{i(<t)}$ and the response completions up to the current token $x_{i(\geq t,<j)}$.

Data Assessment and Selection We aim at finding the most informative subset $S_b \subset S$ from the entire set S under the given budget $|S_b| \leq b$. Mathematically, the selection of S_b requires: 1) a quantitative evaluation function $q(\cdot)$ that assesses each datapoint x_i, and 2) an elaborated sampling mechanism π that determines the rules of selection:

$$S_b = \pi(S, b, q). \tag{2}$$

With respect to the detailed implementation of π, either an iterative, greedy algorithm or a batch-wise heuristic rule can be adopted for compatibility with $q(\cdot)$. For example, the greedy sampling is represented as:

$$S_b = \pi_{\text{greedy}}(S, b, q) = \arg \max_{S' \subseteq S, |S'| \leq b} \sum_{x_i \in S'} q(x_i). \tag{3}$$

The greedy algorithm π_{greedy} consistently selects datapoint x_i with the highest score $q(x_i)$ until the budget target b is met. The heuristic sampling algorithm, on the other hand, chooses each datapoint x_i according to a pre-defined rule. For example, the probability derived by the score $q(x_i)$ can be employed as reference:

$$\begin{aligned} S_b = \pi_{\text{prob}}(S, b, q) = \text{Sample}(S, b, p), \\ p(x_i) = \frac{q(x_i)}{\sum_{x_j \in S} q(x_j)}, \end{aligned} \tag{4}$$

where $\text{Sample}(S, b, p)$ performs sampling b times upon the normalized probability distribution over $q(\cdot)$. More advanced sampling techniques π can be developed in accordance with the domains and tasks at hand.

Such data assessment and selection paradigm is expected to bring about the following benefits: 1) the reduction of noise by ignoring those mislabeled, mismatched instruction-response pairs, 2) the re-balance of data distributions by down-sampling those easy, common, and similar examples while up-sampling hard, rare, and unique ones, and 3) the expedition of training in return for efficient optimization of LLMs.

3 Quality-based Selection

In this section, we present methods on quality assessment and selection. Without lose of generality, we present the unified formulation of quality measurement. The evaluation function $q(x_i)$ in quality-based methods can be decomposed into two fine-grained parts: 1) instruction quality $q_I(x_i)$ and 2) response quality $q_R(x_i)$.

$$q(x_i) = f_q(q_I(x_{i<t}), q_R(x_{i\geq t})), \tag{5}$$

where f_q is an aggregation function that combines the instruction and response quality scores either explicitly or implicitly. Specifically, the instruction quality q_I can be further broken down into: 1) clarity q_I^C that measures the ease of understanding the task, 2) accuracy q_I^A that measures how well the instruction aligns with the expected task, and 3) explicitness q_I^E that measures how explicitly the instruction defines the output constraints (e.g., formats and styles). Consequently, we have $q_I(x_{i<t}) = g_I(q_I^C(x_{i<t}), q_I^A(x_{i<t}), q_I^E(x_{i<t}))$ with the aggregation function g_I. Similarly for the measurement of response, its quality q_R can be assessed via: 1) correctness q_R^C that measures whether the response correctly answers the instruction, 2) coherence q_R^H that measures the logical consistency of the response, and 3) pertinence q_R^P that measures the relevance of the response to the instruction. The final response can be judged as $q_R(x_{i\geq t}) = g_R(q_R^C(x_{i<t}), q_R^H(x_{i<t}), q_R^P(x_{i<t}))$ with an aggregation function g_R. It is noted that all the mentioned quality measurement components above are only demonstrative and are not enforced explicitly in the development of existing quality-based methods. Therefore, certain fine-grained aspects might be integrated into one formulation without extra considerations.

3.1 Hand-crafted Indicators

Overview Traditional methods develop hand-crafted indicators to evaluate the data quality in terms of linguistic analysis such as vocabulary, syntax, and inter-sample semantic similarity. Each indicator is manually, empirically designed with prior knowledge on the language, domain, and task of the corpus under investigation. The calculation of each indicator is explicitly defined and does not require training and inference of proxy models or language models. Although the indicators are hand-crafted, deep learning models such as sentence encoders might be leveraged to extract embedding representations for each datapoint x_i. An indicator IND_i can be typically defined as:

$$\begin{aligned} IND_i = f(IND^1(x_i), IND^2(x_i), \\ IND^3(x_i), ...IND^M(x_i)), \end{aligned} \tag{6}$$

where M denotes the total number of indicators and f is the aggregation function which depends on both the instruction task and dataset. One can simply use linear combinations with pre-defined or dynamically adjusted weights. However, meticulous tuning might be needed for the ultimate f. Given the indicators IND_i for each x_i, two intuitive selection methods can be adopted: 1) to filter out datapoints whose indicator

scores are below a threshold; 2) to keep only the samples whose indicator scores rank within a certain range of percentiles. Mathematically, these two selection mechanisms can be respectively represented as:

$$S_\pi = \{x_i | \tau_{\min} < f(x_i) < \tau_{\max}, 1 \leq i \leq N\}, \tag{7}$$

$$S_\pi = \{x_i | P_{\min} \leq \hat{F}_f(f(x_i)) \leq P_{\max}, 1 \leq i \leq N\}, \tag{8}$$

where $\tau_{\min}$ and $\tau_{\max}$ respectively denote the left and right threshold boundaries. The estimated $\hat{F}_f$ is the empirical cumulative distribution function of all indicators f. $P_{\min}$ and $P_{\max}$ respectively refer to the minimum and maximum percentile for enclosing the selection range. In practice, both the threshold and percentiles are hyper-parameters that require task-specific fine-tuning.

Technical Details [Mishra et al. (2020a)] and [Mishra et al. (2020b)] introduce a data quality metric, namely the DQI, to quantify the differences between successive benchmarks by giving high scores to generalizable samples and low scores to biased samples. Such a metric implies whether a well-trained model truly learns the underlying task rather than overfitting the spurious bias of specific benchmarks. Specifically, DQI has seven components including vocabulary, inter-sample N-gram frequency and relation, inter-sample semantic textual similarities (STS), intra-sample word similarity, intra-sample STS, N-Gram frequency per label, and inter-split STS. Based on the DQI, [Mishra & Sachdeva (2020)] propose to prune existing huge NLP datasets and demonstrates that the model trained on only 2% of the SNLI dataset achieves near-equal performance with that on the entire set. It first performs AFLite [Le Bras et al. (2020)], which is detailed in [Sakaguchi et al. (2021)], to keep samples with predictability scores over a threshold and then delete bottom k samples with the lowest DQI scores. [Dang & Verma (2024)] further split DQI components into linguistic indicators and semantic indicators, and validate their respective roles in detecting outliers, noises, and duplications. Apart from data selection for training LLMs, quality indicators can also be employed to identify the most discriminative samples in the evaluation set to expedite evaluation of LLMs. [Saranathan et al.] investigate key indicators such as spelling errors [Yannakoudakis & Fawthrop (1983)], average word length, excessive word repetition, and the compound probability distribution. These indicators stem from the traditional studies on text readability (i.e., readability formulas and sophisticated features) [Klare et al. (1963; 1984); Dubay (2004); Kintsch & Vipond (2014); Kemper (1983)]. Recent studies on readability leverage NLP systems to extract more advanced and informative features for readability measures [Si & Callan (2001); Collins-Thompson & Callan (2005); Schwarm & Ostendorf (2005); Feng et al. (2010)]. [François (2010; 2011); François & Fairon (2012)] systematically analyze the lexical features, syntactic features, semantic features, and language-specific features with up to 46 indicators. [François & Miltsakaki (2012)] validate these manually-designed (classical) and NLP-enabled (non-classical) readability formulas, implying that high-quality texts can be pinpointed by such carefully designed metrics. [Felice & Specia (2012)] find that the hand-crafted linguistic features should be combined with other shallow features for better quality estimation.

Remark The hand-crafted indicators often stem from studies on linguistic analysis and readability measurement. Although these indicators help filter out instruction samples that are unreadable, nonsensical, and incoherent, they cannot detect mismatched instruction-response pairs and therefore fail to guarantee the instruction-following capability of LLMs trained on highly-scored datasets.

3.2 Model-based Indicators

Overview The model-based indicators, on the other hand, leverage trainable models to predict the indicators for each datapoint. The trainable models used for data quality measurement can either share the same or similar architecture with the language model under development, or possess completely different implementations. Accordingly, these indicators can be simply defined as:

$$\begin{aligned} IND_i = f(&IND^1_{\theta_1}(x_i), IND^2_{\theta_2}(x_i), \\ &IND^3_{\theta_3}(x_i), ...IND^M_{\theta_M}(x_i)), \end{aligned} \tag{9}$$

where the learnable parameters θ_1, θ_2, ..., θ_M highlight the difference between model-based and hand-crafted indicators. Based on such indicators, similar selection mechanisms (Eqs. 7 and 8) can be adopted.

Technical Details One of the most intuitive model-based indicators is perplexity [Shannon (2001); Jelinek et al. (1977); Jelinek (1980)]. It is frequently mentioned as the evaluation metric for pre-trained language models [Penedo et al. (2023); Radford et al. (2018; 2019); Brown et al. (2020); Achiam et al. (2023)] but can also be employed as a data quality indicator. [Ankner et al. (2024)] propose to use a small GPT-style reference model such as MPT 125M [Team (2023)] to prune datasets via perplexity-based sampling for training a 3B model. For any datapoint x_i, the perplexity is defined as the exponential of negative likelihood with base of 2:

$$\begin{aligned} NLL_i &= \frac{1}{|x_i|}\sum_{j=1}^{|x_i|} -\log P(x_{i(j)}|x_{i(<j)};\theta) \\ PPLX_i &= 2^{NLL_i} \end{aligned} \tag{10}$$

Based on the perplexity inferred from a small model, samples at the high and medium percentiles are chosen by Eq. 8 for downstream fine-tuning. [Deng et al. (2021)] develop a unified evaluator framework to score the generated outputs for natural language generation tasks. A RoBERTa-based [Liu et al. (2019)] discriminator learns to score responses in terms of consistency, relevance, preservation, engagingness, and groundedness. One could simply adopt such a discriminator for evaluation of instruction-response pairs. [Zhong et al. (2022)] further propose a multi-dimensional scoring evaluator. For each evaluation dimension, the original instruction-response pairs are converted into positive samples in the form of boolean question-answer problems. The negative samples are respectively constructed via a rule-based transformation. The evaluator itself is implemented as a T5 model [Raffel et al. (2020)] and trained on these positive and negative samples for scoring in the range from 0 to 1. [Jiang et al. (2024c)] prune the UltraChat [Ding et al. (2023)] dataset by scoring each datapoint by the learning complexity of a pre-trained Qwen-1.8B model [Bai et al. (2023)]. Specifically, the learning complexity is calculated as the averaged prediction confidence of different subnets:

$$\tilde{S}(x_i) = \frac{1}{I}\sum_{j=1}^{I} PPLX_{i;\Theta_j}, \tag{11}$$

where I is the number of subnets. Each subnet Θ_j is obtained by adjusting the dropout rate from 10% to 90% on the original Θ of any pre-trained language model. Such a dropout technique imitates the learning progress in a training-free manner. Datapoints with small $\tilde{S}(x_i)$ are considered as easy ones. They are often learned at an earlier training stage with less model capacity compared with hard ones. In a data-poor regime, easy datapoints are more informative and should be kept first. On the contrary, in a data-rich regime, hard datapoints should be treasured in that more samples can be preserved for fine-tuning. Both [Bukharin & Zhao (2023)] and [Du et al. (2023)] employ reward models to assess the quality of each instruction pairs. They respectively utilize the raft model [Dong et al. (2023)] and the deberta-v3-large-v2 [1] for reward scoring:

$$R_i = r_\theta(x_{i(<t)}, x_{i(\geq t)}), \tag{12}$$

where r_θ denotes the reward model. t is the index where $x_{i(<t)}$ and $x_{i(\geq t)}$ respectively denote the instruction Q and response A. [Marion et al. (2023)] investigate three classic metrics in clean set selection [Guo et al. (2022); Song et al. (2022); Natarajan et al. (2013); Qin et al. (2024)]: perplexity (Eq. 10), error l_2-Norm (EL2N) [Paul et al. (2021)], and memorization ranking [Biderman et al. (2024)]. Specifically, EL2N is defined as:

$$EL2N_i = \frac{1}{|x_i|}\sum_{j=1}^{|x_i|} \|P(x_{i(<j)};\theta) - \mathbf{y}_{i(j)}\|_2, \tag{13}$$

where $\mathbf{y}_{i(j)}$ denotes the one-hot ground-truth vector as the target of the probability vector $P(x_{i(<j)};\theta)$. Specifically, both $P(x_{i(<j)};\theta) \in \mathbb{R}^{N_{\text{vocab}}}$ and $\mathbf{y}_{i(j)} \in \mathbb{R}^{N_{\text{vocab}}}$ are of the same dimension, where N_{vocab} denotes the vocabulary size associated with the tokenizer. For the vector of $\mathbf{y}_{i(j)}$, all elements are zero except that the element indexed at $x_{i(j)}$ is one. The memorization ranking is represented as:

$$MEM_i = \frac{1}{N_{\text{win}}}\sum_{j=1}^{N_{\text{win}}} \mathbb{1}(\hat{x}_{i(M_{\text{offset}}+j)} = x_{i(M_{\text{offset}}+j)}), \tag{14}$$

[1] https://huggingface.co/OpenAssistant/reward-model-deberta-v3-large-v2

where N_{win} denotes the length of a consecutive sequence and M_{offset} is an offset of the starting index. The $\hat{x}_{i(M_{\text{offset}}+j)}$ refers to the generated token given input $x_{i(<M_{\text{offset}}+j)}$, and $x_{i(M_{\text{offset}}+j)}$ is its ground-truth. [Cao et al. (2023)] combine both hand-crafted indicators (e.g., input length, output length, MTLD [McCarthy & Jarvis (2010)], and kNN-i [Dong et al. (2011)]) and model-based indicators (e.g., reward score, perplexity, and Uni-Eval metrics [Zhong et al. (2022)]) for fitting the loss of a LLM on the evaluation set. The linear regression model is optimized via least squares method [Bjork (1988)] and the optimal selection of instruction data is achieved via BlendSearch [Wang et al. (2021a;b)] for minimizing the estimated evaluation loss. [Li et al. (2023a)] propose one of the most pioneering works that leverages the target language model itself to perform self-guided data selection. The language model is first "warmed-up" with very few samples randomly chosen from the pool to learn from brief experience. Then, such an experienced model evaluates each instruction-response pair via the instruction-following difficulty (IFD) score. The IFD score measures how much guidance or assistance the instruction provides to the generation of ground-truth response, by comparing the loss of causal language modeling on the response with and without instruction:

$$\begin{aligned} IFD_i &= \frac{NLL_i^{A|Q}}{NLL_i^{A}}, \\ NLL_i^{A|Q} &= \frac{1}{|x_{i(\geq t)}|}\sum_{j=t}^{|x_i|} -\text{log}P(x_{i(j)}|x_{i(<j)};\theta), \\ NLL_i^{A} &= \frac{1}{|x_{i(\geq t)}|}\sum_{j=t}^{|x_i|} -\text{log}P(x_{i(j)}|x_{i(t\leq,<j)};\theta), \end{aligned} \tag{15}$$

where the index t splits apart the instruction Q and the response A. Samples whose IFD scores over $\tau_{\max} = 1$ are invalid datapoints with misaligned, mismatched instruction-response pairs. The empirical setting of $\tau_{\min}$ affects the trade-off between quality and diversity of the selected datapoints. [Zhao & Fang (2024)] comprehensively employ hand-crafted indicators for low-level quality filtering, and uses perplexity and IFD score for high-level filtering. A voting mechanism is additionally introduced with IFD scores from one pre-trained base model and one fine-tuned experience model. [Li et al. (2024b)] corroborate that both the perplexity and IFD scores inferred from a rather small GPT2-125M [Radford et al. (2019)] are indicative in selecting high-quality datapoints for training LLaMA2-7B and LLaMA2-13B [Touvron et al. (2023b)], which greatly improves selection efficiency.

Another popular model-based quality filtering method is AF-Lite [Le Bras et al. (2020)], which has been applied and validated in recent NLP studies [Mishra & Sachdeva (2020); Sakaguchi et al. (2021)]. It randomly partition all available datapoints into the training set and the validation set. Then, a model (e.g., linear classifier or language model) is trained on the training set and inferred on the validation set. Such process iterates m times for calculation of the predictability score, which is defined as the ratio of the number of correctly predicted response over the number of total predictions:

$$\begin{aligned} PRED_i &= \frac{|\{\hat{x}_i \in E_i,\ s.t.\ \hat{x}_i = x_i\}|}{E_i}, \\ E_i &= \{\hat{x}_i^{\theta_1}, \hat{x}_i^{\theta_1}, ..., \hat{x}_i^{\theta_j}, ..., \hat{x}_i^{\theta_m}\}, \end{aligned} \tag{16}$$

where $\hat{x}_i^{\theta_j}$ denotes the generated response from the model parameterized as θ_j. It is noted that x_i is not involved for optimizing θ_j, and therefore a higher $PRED_i$ suggests better quality.

[Bhatt et al. (2024)] present uncertainty-based indicators such as mean entropy [Settles (2011); Kremer et al. (2014)], least confidence [Settles (1995; 2011)], mean margin [Tong & Koller (2001); Balcan et al. (2006); Settles (2011)], and min margin [Nguyen et al. (2022)]. Mathematically, such uncertainty indicators are defined as:

$$\begin{aligned} U_i^{\text{entropy}} = \frac{1}{|x_i|}\sum_{j=1}^{|x_i|} P(x_{i(j)}|x_{i(<j)};\theta)\cdot \\ \text{log}P(x_{i(j)}|x_{i(<j)};\theta). \end{aligned} \tag{17}$$

$$U_i^{\text{confidence}} = -\prod_{j=1}^{|x_i|} P(x_{i(j)}|x_{i(<j)};\theta). \tag{18}$$

$$U_i^{\text{margin}} = -\frac{1}{|x_i|}\sum_{j=1}^{|x_i|}(\beta_1(P(x_{i(<j)};\theta)) - \beta_2(P(x_{i(<j)};\theta))), \tag{19}$$

$$U_i^{\text{min-margin}} = -\min_{j\in\{1,2,\ldots,|x_i|\}}(\beta_1(P(x_{i(<j)};\theta)) - \beta_2(P(x_{i(<j)};\theta))), \tag{20}$$

where β_1 and β_2 denote the largest and second largest elements of the probability $P(x_{i(<j)};\theta) \in \mathbb{R}^{N_{\text{vocab}}}$ for the newly generated j-th token. However, [Wu et al. (2023)] find that such uncertainty-based data sampling methods perform worse than random sampling on Databricks-Dolly [Conover et al. (2023)], SelfInstruct-Davinci [Taori et al. (2023)], and SelfInstruct-GPT4 [Peng et al. (2023)].

Remark Hybrid techniques that simultaneously combine perplexity, uncertainty, reward scores, and other training-aware metrics are promising in selecting unbiased high quality samples. In consideration of the training and inference cost, it is feasible to employ small proxy models as alternatives for computing model-based indicators.

3.3 GPT Score

Overview The invoking of OpenAI APIs [Tingiris & Kinsella (2021); Lappalainen & Narayanan (2023); Sun et al. (2023); Kublik & Saboo (2023)] for ChatGPT services (e.g., GPT3.5, GPT4) allows automatic scoring of instruction tuning datasets. Recent studies on bringing LLMs as judges [Zheng et al. (2024); Wang et al. (2023a); Zhu et al. (2023); Huang et al. (2024); Zeng et al. (2023); Chan et al. (2023)] reveal that powerful language models like ChatGPT highly align with human preference on judging the quality of instructions and responses. Given a well-designed prompt with clear definition on grading criteria, ChatGPT produces justified quality scorings with explanations for the raw textual instruction data I_i:

$$GPTScore_i = G(I_i, p_G), \tag{21}$$

where p_G denotes the prompt template that defines the task and grading scheme with format constraints on outputs (see Fig. 4). The $G(\cdot,\cdot)$ represents the process of quality scoring and response parsing, which produces the GPT score $GPTScore_i$. Samples with high $GPTScore_i$ can be selected using Eqs. 7 and 8.

Prompt p_G for scoring I_i with instruction (input) and response in the <dimension>

We would like to request your feedback on the performance of AI assistant in response to the instruction and the given input displayed following.

Instruction: <instruction>
Input: <input>
Response: <response>

Please rate according to the <dimension> of the response to the instruction and the input. Each assistant receives a score on a scale of 0 to 5, where a higher score indicates higher level of the <dimension>. Please first output a single line containing the value indicating the scores. In the subsequent line, please provide a comprehensive explanation of your evaluation, avoiding any potential bias.

Figure 4: The prompt p_G for scoring the raw text I_i with ChatGPT.

Technical Details [Chen et al. (2023b)] propose a surprisingly easy-yet-effective method that directly uses GPT3.5 to score datapoints in terms of helpfulness and accuracy. Both instructions and responses are scored on a scale from 0 to 5 and experimental results show that general instruction datasets, except coding-related samples, can be distilled into smaller subsets for better downstream performance. [Bukharin & Zhao (2023)] follow [Chen et al. (2023b)] for filtering Alpaca [Taori et al. (2023)]. [Chen & Mueller (2024)] employ the BSDetector [Chen & Mueller (2023)] to estimate the confidence of GPT3.5/GPT4 on the give instruction-response pair. It takes both the self-consistency and direct scoring into consideration. Only highly confident samples are kept for fine-tuning domain-specific LLMs and those less confident ones are corrected automatically by these LLMs. [Xu et al. (2023b)] directly evaluate instruction datasets in terms of accuracy, explanation, clarity, and difficulty for weighted scorings from GPT4. Then, both hand-crafted indicators (i.e., lengthwise semantic evaluation) and GPT4 scorings are employed for final ranking. [Liu et al. (2023b)] argue that the direct scoring of GPT4 on one single instruction sample is not well-calibrated and instead gives relative ranking of multiple instruction variants at once. The complexity of instructions [Xu et al. (2023a)] and the quality of instruction-response pairs are sequentially obtained from GPT3.5. [Zhang et al. (2024c)] use GPT scorings to judge: 1) whether the given text contains mathematical contents; 2) and if yes, whether these maths contents are of high quality for education purpose. Such scores are proved more effective than traditional "mathematical" classifiers [Paster et al. (2023)]. [Lu et al. (2023a)] propose to use ChatGPT for annotating open-ended, fine-grained intention tags on open datasets. Then, the quality of the tag dataset is evaluated by humans and GPT4 in terms of tagging precision and consistency. Instead of fully relying on the GPT4, [Li et al. (2023c)] exploit the model under investigation itself (e.g., LLaMA 65B) to iteratively derive quality scores on each augmented example on a 5-point scale. Then a curated clean set is chosen via the Eq. 7. QuRator [Wettig et al. (2024)] manually define quality criterion such as writing style, facts and trivia, educational value, and required expertise. Then, quality comparison is conducted on two instruction-response samples via GPT3.5 scoring. Such pairwise scorings are used to fine-tune a sheared-LLaMA 1.3B model [Xia et al. (2023)] in a manner similar to DPO [Ouyang et al. (2022); Rafailov et al. (2024)]. It is noted that the pairwise scoring [Ouyang et al. (2022); Dubois et al. (2024); Zeng et al. (2023); Liu et al. (2023b)] has been found more reliable, consistent, and unbiased than the individual scoring [Gunasekar et al. (2023); Chen et al. (2023b)] during GPT-based quality analysis.

Remark Closed-source LLMs such as ChatGPT enjoy a high level of alignment with human preference and therefore can be utilized for quality scoring. It would be cost-efficient to collect few (e.g., <100K) GPT-scored samples first and then fine-tune an open-source LLM for quality measurement on massive corpus.

3.4 Human Evaluation

Overview Evaluation with human-in-the-loop is indispensable in constructing preference alignment datasets [Wang et al. (2023b); Ouyang et al. (2022)] for helpfulness, honesty, and harmlessness. Specifically, human annotators deliver gradings following specific criteria (see Fig. 5) in multiple dimensions:

$$\begin{aligned} LabelScore_i = f(&LabelScore^1(x_i), \\ &LabelScore^2(x_i), ..., LabelScore^M(x_i)), \end{aligned} \tag{22}$$

where $LabelScore^m(x_i)$ can be both bool or integer (e.g., range from 0 to 5) for the m-th fine-grained aspect. The aggregation function f is commonly chosen as summation or averaging.

Technical Details The OpenAssistant [Köpf et al. (2024)] dataset is featured by its high-quality human-generated, human-annotated multi-lingual conversations for both instruction tuning and reinforcement learning from human feedback. For each instruction-response pair along the conversation tree, the human annotators are asked to categorize them according to three dimensions: spam detection, guideline adherence, and quality. The quality score is rated on a five-point *Likert* scale across aspects including quality, creativity, humorousness, politeness, and harmlessness. These scores are used to sort instructions for analysis and preference optimization of LLMs. [Lu et al. (2023a)] enroll human annotators to provide judgements on the tagging of each instruction. To verify the quality scores provided by humans, counterfactual cases are prepared respectively for precision and consistency tasks. Results show that human annotators have low false

Guidelines (excerpts) for human annotations

Guidelines
Below is a list of guidelines that should be adhered to for each possible task available when building the dataset. To see some examples of how the guidelines can be applied, visit the examples document.

1. General rules
- Always make sure to read and understand the guidelines to each task before fulfilling it. - Try to follow the guidelines as closely as possible. - If you are unsure whether a message violates a guidelines, contact us at our Discord.
- Use the thumbs-up/thumbs-down system to further mark messages that are of high or low quality.

2. Providing an assistant reply #assistant-reply
Do:
- Remain polite and treat the user with respect, even when not given the same courtesy.
...

Figure 5: The guidelines (thumbnails) for human experts to create and annotate instruction datasets.

positive rates at tagging precision, but lack proof of confidence on their original quality judgements. [Zhou et al. (2024a)] propose to use human annotators for creation of small-yet-effective instruction datasets. To collect questions and answers from various sources, simple hand-crafted indicators such as text length are used to filter low-quality datapoints. Then, high quality instruction-response pairs are manually selected (750) and written (250) via subjective quality control. The databricks-dolly dataset [Conover et al. (2023)] contains 15K human-generated instruction-response pairs. Although quality is emphasized during large-scale annotation, imperfect samples still exist. For example, low-quality and inaccurate responses, incomplete and vague instructions, problematic texts with toxic language and grammar errors are found [He et al. (2024)].

Remark Human evaluation plays an irreplaceable role in quality control of preference alignment. To reduce the inter-annotator inconsistency, detailed guidelines should be prepared for quality measurement. In addition, supplementary quality measures such as GPT scores can be provided for reference during evaluation and selection of high-quality datapoints.

4 Diversity-based Selection

In this section, we introduce methods that emphasize the diversity of instruction datasets. When it comes to diversity, existing researches either measure the individual diversity of each sample (e.g., lexical and semantic richness) or the overall diversity of the entire dataset (e.g., the volume of the enclosed embedding space). Datapoints whose tasks and domains are of minority classes in a long-tailed distribution are preferred during subset selection. Such sampling philosophy strikes to maintain or approximate the spread of the original embedding clusters but with much less sparsity. In the case of diversity, the evaluation function $q(x_i)$ can be represented in the unified formulation as:

$$q(x_i) = f_d(q_L(x_i), q_S(x_i)), \tag{23}$$

where q_L measures the lexical diversity of x_i and q_S assesses the semantic diversity. The f_d denotes the aggregation function in diversity measurement. Typically, q_L often investigates the diversity of n-grams, tokens, words, and sequences. Complementarily, q_S emphasizes semantic diversity that the variety of representations of the selected datapoints should be maximized in the embedding space. Both the two aspects of diversity can be sequentially or jointly considered to remove any duplicates in the instruction datasets.

4.1 Hand-crafted Indicators

Overview The diversity of datasets is the key to develop less biased, more generalizable machine learning models. However, recent studies [Zhao et al. (2024c;b)] show that existing datasets do not share a unified

and concrete definition of diversity in terms of dataset composition, source, domain, subject, annotator, and promote (fairness). With respect to the diversity measures specific in instruction tuning datasets, hand-crafted indicators, similar to Eq. 6 in traditional NLP studies, can be used as a good starting point.

Technical Details One of the most popular diversity measure is lexical diversity, which refers to the range of different words occurring in one text. The greater range implies greater diversity and quality. Type-token ratio (TTR) [Templin (1957); Richards (1987)] is originally proposed as:

$$TTR_i = \frac{|Unique(x_i)|}{|x_i|}, \tag{24}$$

where $Unique(x_i)$ denotes the set of unique tokens present in x_i. To reduce the sensitivity of TTR to the variation of text length, several studies [Covington & McFall (2008; 2010); Kettunen (2014); Matlach et al. (2021)] standardized the length by introducing logarithms or n-grams into the formula.

Later, computational approaches to measure lexical diversity have been developed such as vocabulary diversity (vocd-D) [Malvern & Richards (1997); Malvern et al. (2004); Silverman & Ratner (2002); deBoer (2014)], the measure of textual lexical diversity (MTLD) [McCarthy & Jarvis (2010); Jarvis & Daller (2013)], and hypergeometric distribution diversity (HD-D) [Jarvis (2013); McCarthy (2005)]. All these metrics require multi-step computation for approximation. Specifically for vocd-D, random sampling is first performed on x_i for a series of sub-sequences with varying lengths k (e.g., 10, 20, 30 tokens). Then, TTR^k is:

$$TTR_i^k = \frac{|Unique(x_{i(j\leq,<j+k)})|}{|x_{i(j\leq,<j+k)}|},\ 1 \leq j \leq |x_i| - k, \tag{25}$$

where $x_{i(j\leq,<j+k))}$ denotes the sub-sequence of x_i starting from the randomly chosen index j and ending at the index $j+k$. Then, the curve of TTR_i^k versus the lengths k is plotted and a mathematical model is built for fitting the curve:

$$\hat{TTR}_i^k = \frac{\mathcal{D}}{k}[(1 + 2\frac{k}{\mathcal{D}})^{\frac{1}{2}} - 1], \tag{26}$$

where $\mathcal{D}$ is the only parameter required to be estimated. By approximating $\hat{TTR}_i^k$ towards TTR_i^k with the least squares, we have $\mathcal{D}_{\text{best fit}} = D$:

$$\textit{vocd-D}_i = D. \tag{27}$$

A larger D reflects the higher diversity of x_i. The computation of MTLD, on the other hand, first determines the TTR_i as a pre-defined threshold, and then partitions x_i into M different contiguous subsequences $\{x_i^1, x_i^2, ..., x_i^m, ..., x_i^M\}$. Each subsequence $x_i^m = x_{i(j\leq,<j+k)}, \forall k > 0, \forall 1 \leq j \leq |x_i| - k$ maintains a TTR_i^k above the threshold TTR_i. The MTLD is defined as:

$$MTLD_i = \frac{1}{M}\sum_{i=1}^{M}|x_i^m|. \tag{28}$$

The HD-D shares the same idea behind vocd-D but stems from the hypergeometric distribution [McCarthy & Jarvis (2010)]. With M-times sampling, the HD-D represents the probability of drawing a certain number of tokens of the given type from the subsequence of x_i with a particular size k:

$$\begin{aligned} \textit{HD-D}_i = & \sum_{t=1}^{|Unique(x_i)|} \frac{1}{M}\sum_{m=1}^{M} \mathbb{1}(x_{i(n)}^m = u_t), 1 \leq n \leq |x_i^m|, \\ & u_t \in Unique(x_i), x_i^m = x_{i(j\leq,<j+k)}, \forall k > 0, \forall 1 \leq j \leq |x_i| - k. \end{aligned} \tag{29}$$

Variants of TTR indicators such as MTTRSS [Malvern et al. (2004)], MSTTR [Malvern et al. (2004)], MATTR [Covington & McFall (2010)], and MTLD-W [Vidal & Jarvis (2020); Kyle et al. (2021)] all target at the solutions to two fundamental problems [Bestgen (2023)]: 1) the sensitivity of indicators to text length, and 2) the impact of the indicator parameters. [Li et al. (2015)] propose two rather simplified TTR scores as

distinct-1 and *distinct*-2, where the number of distinct unigrams and bigrams of x_i are respectively divided by the total number of tokens. Other studies [Cao & Clark (2017); Zhu et al. (2018); Shu et al. (2019); Tevet & Berant (2020)] extend the application of n-gram-based diversity for model-generated responses.

Apart from lexical diversity, there exists many efficient diversity indicators that are built upon the semantics of each example. [Dong et al. (2011)] propose to approximate k-nearest neighbor (k-NN) graph [Peterson (2009)] with arbitrary similarity measures on semantic embeddings of large-scale datasets. Such efficient construction of a k-NN graph allows the distance of x_i to its j-th nearest neighbors to be a feasible diversity measure:

$$kNN_i^j = d(g(x_i), g(N_j(x_i))), \tag{30}$$

where $N_j(x_i)$ denotes the j-th closest neighbor of x_i in the embedding space projected by $g(\cdot)$. The common choices of the distance function $d(\cdot,\cdot)$ include the Euclidean distance, cosine distance, and Jaccard coefficient distance [Huang et al. (2008)]. The projection from text (e.g., instruction-response pairs) into the embedding space can be achieved with pre-trained sentence BERT [Reimers & Gurevych (2019); Feng et al. (2020)], where an additional pooling operation is performed on the final output of BERT [Devlin et al. (2018)] for sentence embeddings. Note that a higher kNN_i implies that the sample x_i is more unique and should be kept in subset selection for higher diversity. Due to the fine-grained representation capability of BERT, existing hand-crafted indicators often rely on BERT embeddings for similarity or diversity measurement [Tevet & Berant (2020); Zhang et al. (2019); Larson et al. (2019); Yauney et al. (2023)].

To improve the generalization of diversity measure, [Xu et al. (2023b)] argue that the statistics of feature embedding of each sample itself should be considered. It does not require additional prior knowledge on the structure of embeddings. Given all datapoints $x_i \in S$, their semantic embeddings from any sentence encoder can be represented as $X = [g(x_1), g(x_2), ..., g(x_N)] \in \mathcal{R}^{|S|\times H}$. The row variance Var_i of each embedding $g(x_i)$ in the reduced dimensional space $\mathcal{R}^{|S|\times k}$ by principal components analysis (PCA) [Wold et al. (1987)] is used as the diversity indicator:

$$\begin{aligned} Var_i &= \frac{1}{k-1}\sum (j=1)^k (Y_{ij} - \mu_i)^2, \\ \mu_i &= \frac{1}{k}\sum_{j=1}^{k} Y_{ij} \end{aligned} \tag{31}$$

where the PCA chooses the top-k eigenvectors ($V = [v_1, v_2, ..., v_k]$ with $\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_k$) of the covariance matrix $Cov = Q\Lambda Q^T = \frac{1}{|S|-1}(X-\mu_X)^T(X-\mu_X),\ \mu_X = \frac{1}{|S|}\sum_{i=1}^{|S|} X_i$ to project the original embeddings into more compact and reduced ones via $Y = (X-\mu_X)V$. Samples with the highest 20% Var_i (via Eq. 8) are selected as the variety-curated dataset.

When it comes to the overall diversity of a dataset S, the average distance of any datapoint x_i to its closest neighbor in the dataset, namely kNN_i, can be leveraged intuitively:

$$D^{kNN}(S) = \frac{1}{|S|}\sum_{i=1}^{|S|} kNN_i^1,\ x_i \in S. \tag{32}$$

Such a diversity measure has been widely used in dataset construction and retrieval [Stasaski et al. (2020); Stasaski & Hearst (2022); Mithun et al. (2019); Spyromitros-Xioufis et al. (2015); Sun et al. (2024a); Ionescu et al. (2018)]. [Du & Black (2019)] simply perform clustering on all samples with k-means [Ikotun et al. (2023)] into K clusters (C_1,C_2,...,C_K) in the embedding space, and then uses the cluster inertia as diversity indicators:

$$\begin{aligned} D^{inertia}(S) &= \sum_{j=1}^{K}\sum_{x_i\in C_j} \|g(x_i) - \mu_j\|^2, \\ \mu_j &= \frac{1}{|C_j|}\sum_{x_i\in C_j} g(x_i). \end{aligned} \tag{33}$$

[Lai et al. (2020)] develop a diversity metric on the dispersion of a cluster induced by embeddings of all samples, where the cluster is approximated by a multi-variate Gaussian distribution:

$$D^{radius}(S) = \sqrt[H]{\prod_{j=1}^{H} \sigma_j}, \tag{34}$$

where H is the dimension of the projected embeddings $g(x_i) \in \mathcal{R}^H$ and σ_j denotes the radius of the ellipsoid along the j-th axis of the dataset S. The inter-cluster (class) distance can also be used for diversity measure [Dang & Verma (2024)]:

$$D^{ICD}(S) = \frac{1}{K} \sum_{j=1}^{K} \text{div}_{JS}(P_j || P_{\neq j}), \tag{35}$$

where P_j denotes the inverse-document frequency (IDF) distribution [Sparck Jones (1972)] of the cluster C_j and div_{JS} is the Jensen-Shannon divergence.

Remark Both lexical and semantic diversity should be considered with hand-crafted indicators. The optimization of individual diversity would contribute to the overall diversity of the entire dataset.

4.2 Model-based Indicators

Overview Similar to Eq. 9, model-based indicators on diversity also rely on the target or proxy language model for computing the indices.

Technical Details The diversity of a dataset S can be intuitively defined as the sum of rarity measures of each constituting element x_i. Accordingly, entropy-related methods are proposed to estimate such rarity. The more uncommon, various samples exist, the higher diversity the dataset becomes. Mathematically, the vanilla entropy [Shannon (1948)] is proposed for diversity measures:

$$D^{entropy}(S) = -\sum_{x_i \in S} P(x_i|\theta) \cdot \log_2(P(x_i|\theta)), \tag{36}$$

where $P(x_i)$ denotes the probability of x_i occurring in the dataset. Later, Rényi entropy [Rényi (1961)] introduce an additional parameter $\alpha > 0, \alpha \neq 1$ for a generalized entropy definition:

$$D_{\alpha}^{RE}(S) = \frac{1}{1-\alpha} \log_2(\sum_{x_i \in S} P(x_i|\theta)^{\alpha}). \tag{37}$$

The parameter α adjusts the element-wise emphasis on rare or frequent events.

Studies on biology and ecology [Mouillot & Lepretre (1999); Peet (1974); He & Hu (2005); Gregorius & Gillet (2008)] investigate Simpson's Index (SI) [Simpson (1949); Wu et al. (2024; 2022)] for measuring the biodiversity of species and genetics. [Zhou et al. (2020)] propose a variant of the original SI with a more flexible statistic metric:

$$D^{SI}(S) = 2\frac{\sum_{x_i, x_j \in S, i \leq j} \mathbb{1}(x_i = x_j|\theta)}{|S|(|S|+1)}, \tag{38}$$

where the equivalence of x_i and x_j is judged by an indicator function parameterized as θ.

Vendi Score (VS) [Dan Friedman & Dieng (2023); Pasarkar & Dieng (2023); Nguyen & Dieng (2024)] is rencently proposed for diversity measurement in machine learning researches. Inspired by the Rényi entropy, a generalized VS metric [Pasarkar & Dieng (2023)] is defined as below:

$$D_{\alpha}^{VS}(S) = \exp(\frac{1}{1-\alpha} \log_2(\sum_{i=1, i \in supp(\bar{\lambda})}^{|S|} \bar{\lambda}_{i|\theta}^{\alpha})), \tag{39}$$

where $\bar{\lambda}_{i|\theta}$ denotes the normalized eigenvalues of the similarity kernel matrix $K_{S|\theta}$, and $supp(\bar{\lambda})$ is the set of indices of all non-zero eigenvalues. The smaller $\alpha < 1$ makes the scoring more sensitive to rare classes and therefore allows accurate diversity measurement even under severe class imbalance. One simple implementation of the similarity kernel $K_{S|\theta}$ is to use the Gaussian Radial Basis function k with feature embeddings as $k(g(x_i|\theta), g(x_j|\theta)) = \exp(-\frac{1}{2}\|g(x_i|\theta) - g(x_j|\theta)\|^2)$. [Nguyen & Dieng (2024)] further introduce quality scoring into Eq. 39 where for each subset $S_b \subset S$, its average quality score $Q(S_b) = \frac{1}{|S_b|}\sum_{x_i \in S_b} IND_i$ is multiplied with $D_\alpha^{VS}(S_b)$ for comprehensive evaluation in terms of quality and diversity.

[Miranda et al. (2022)] propose an intrinsic diversity coefficient to measure the diversity of a dataset with Task2Vec embeddings [Achille et al. (2019); Nguyen et al. (2019)] for distance computation between different tasks. The Task2Vec encodes data from different tasks by the diagonal entries of the Fisher Information Matrix (FIM). The FIM results from fine-tuning only the final (e.g., token classification) layer of a pre-trained model, namely a probe model (e.g., GPT2 [Radford et al. (2019)]), to solve the task. Given a batch of samples B, the mathematical representation of FIM is defined as:

$$\begin{aligned} \hat{F}_B = \mathbb{E}_{x_i, j, \hat{x}_{i(j)}} \nabla_\theta \log P(\hat{x}_{i(j)}|x_{i(<j)};\theta) \cdot \\ \nabla_\theta \log P(\hat{x}_{i(j)}|x_{i(<j)};\theta)^T, \end{aligned} \tag{40}$$

where $\hat{x}_{i(j)}$ denotes the j-th token predicted from the model parameterized as θ given the real sequence input $x_{i(<j)}$. The expectation $\mathbb{E}_{x_i, j, \hat{x}_{i(j)}}$ takes an average over the sequence length $|x_i|$ for each x_i sampled randomly from the batch $x_i \in B$. The Task2Vec embedding $\vec{f_B} = diag(F_B)$, where $diag(\cdot)$ denotes the diagonal entries of F_B. Based on the Task2Vec embeddings, [Lee et al. (2023)] propose to compute the diversity coefficients $\hat{div}$ specifically for NLP datasets:

$$\begin{aligned} D^{\hat{div}}(S) = \mathbb{E}_{B_1, B_2 \sim S} d(\vec{f_{B1}}, \vec{f_{B2}}), \\ D^{\hat{div}}(S_1, S_2) = \mathbb{E}_{B_1 \sim S_1, B_2 \sim S_2} d(\vec{f_{B1}}, \vec{f_{B2}}), \end{aligned} \tag{41}$$

where d denotes distance measurement (e.g., cosine distance). Both B_1 and B_2 are two batches sampled respectively from the same or different datasets for diversity measures within or across datasets. Experiments confirm that hand-crafted indicators such as the number of latent concepts [Xie et al. (2021)] and the richness of vocabulary are positively associated with the proposed $\hat{div}$ coefficients.

[Lu et al. (2023a)] develop a diversity measure by open-ended tagging. Specifically, a tagging model parameterized by θ is trained with GPT4-labeled tagging pairs. To label such tags of each datapoint, the GPT4 first performs open-set fine-grained tagging and then all the collected tags are normalized to filter out low-frequency ones and aggregate near-duplicate ones. The specifically trained tagging model describes each instruction tuning datapoint x_i by its fine-grained, atomic intentions and semantics (e.g., tasks and domains). Correspondingly, the number of tags can be viewed as a diversity indicator for sampling a instruction subset S_b from the whole set S (see Alg. 1).

Remark The model-based indicators are highlighted by their flexibility in handling various aspects of diversity either implicitly or explicitly.

4.3 Geometry-based Coreset Sampling

Overview Instead of explicitly calculating the diversity-aware indicators, recent studies on selecting instruction datasets tend to introduce coreset sampling methods for a systematic consideration [Guo et al. (2022)]. Specifically, coreset sampling aims to find the most informative-and-diverse subset that represents the entire dataset the most, so that close or even surpassing performance can be achieved on the language model trained on the subset with respect to that on the entire set.

Technical Details Among different categories of coreset sampling methods, geometry-based methods are the most intuitive and widely-used ones [Chen et al. (2012); Agarwal et al. (2020); Sener & Savarese (2017); Sinha et al. (2020); Kamalov (2020); Rezazadegan Tavakoli et al. (2011); Kirchenbauer et al. (2024);

Algorithm 1 TagLM-based Diverse Sampling [Lu et al. (2023a)]

Require: data $x_i \in S$, a tagging LLM T_θ, a visited tag set D_b^B, and a budget b

Initialize $S_b = \emptyset$
for each $x_i \in S$ **do**
    Obtain tags $D_{x_i} = T_\theta(x_i)$
end for
repeat
    Initialize $D_b^B = \emptyset$
    for each $x_i = \arg\max_{x_i \in S} D_{x_i}$ **do**
        if $|D_b^B \cup D_{x_i}| > |D_b^B|$ **then**
            $S_b = S_b \cup \{x_i\}$
            $D_b^B = D_b^B \cup D_{x_i}$
            $S = S \backslash \{x_i\}$
        end if
    end for
until $|S_b| = b$
return S_b

Zhou et al. (2023)]. The intuition behind is that close samples in the embedding space often share similar properties with low diversity. Therefore, redundant information can be effectively suppressed by controlling the minimum distance between any two samples for subset selection. Specifically, k-center greedy is a typical diversity-oriented sampling method for massive pretraining and instruction-tuning corpus [Chen et al. (2023a); Bhatt et al. (2024); Wu et al. (2023); Zhao & Fang (2024); Du et al. (2023)]. It solves the minimax facility location (FL) problem [Cornuéjols et al. (1983); Farahani & Hekmatfar (2009)], i.e., selecting the subset S_b under the given size budget b from the full set S so that the largest distance between an example in $S \backslash S_b$ and its closest example in S_b is minimized:

$$\min_{S_b \subset S,\ |S_b|=b} \max_{x_i \in S \backslash S_b} \min_{x_j \in S_b} d(g(x_i), g(x_j)). \tag{42}$$

The direct solution to Eq. 42 is NP-hard [Cook et al. (1994)] and a greedy approximation is proposed [Sener & Savarese (2017)] (see Alg. 2). Instead of simultaneously retrieving all the datapoints that can maximize the diversity of the selected subset, k-center greedy iteratively finds the most heterogeneous datapoint until the budget b runs out. For initialization of S_b^0, one can either choose randomly sampled datapoints from S, or use the cluster center points from K clusters $(C_1, C_2, ..., C_K)$ of S via k-means clustering. Similarly, the farthest point sampling method [Eldar et al. (1997)] shares the same principle that each iteration time only the farthest datapoint relative to the already selected coreset is chosen from the candidates.

Algorithm 2 K-Center Greedy [Sener & Savarese (2017)]

Require: data $x_i \in S$, existing pool S_b^0 and a budget b

Initialize $S_b = S_b^0$
repeat
    $u = \arg\max_{x_i \in S \backslash S_b} \min_{x_j \in S_b} d(g(x_i), g(x_j))$
    $S_b = S_b \cup \{u\}$
until $|S_b| = b + |S_b^0|$
return $S_b \backslash S_b^0$

In addition to the k-center greedy, the herding methods [Chen et al. (2012); Welling (2009); Huszár & Duvenaud (2012); Adhikary & Boots (2022)] select datapoints x_i so that the distance between the coreset center and the full set center is minimized in the embedding space. For efficiency, it is also approximated via greedy implementation [Chen et al. (2016); Harvey & Samadi (2014)] by adding one sample each time into the S_b to minimize the distance between two centers (see Alg. 3). Similar to k-center greedy, the herding

greedy also performs iterative sample selection. However, it differs in that in each step, the distance between the centers of the selected set and the full set is minimized.

Algorithm 3 Herding Greedy [Harvey & Samadi (2014)]

Require: data $x_i \in S$, a budget b

1: Initialize $\mu = \frac{1}{n}\sum_{i=1}^{n} g(x_i)$
2: Initialize $S_b = \emptyset$
3: **for** $t = 1$ to b **do**
4: $\quad u = \arg\min_{x_i \in S \setminus S_b} \|\mu - \frac{1}{|S_b|+1}\sum_{x_j \in S_b \cup \{x_i\}} g(x_j)\|^2$
5: $\quad S_b = S_b \cup \{u\}$
6: **end for**
7: **return** S_b

Furthermore, recent studies tend to develop complex heuristic sampling methods that takes geometry-based diversity into consideration [Jiang et al. (2023c); Chan et al. (2021); Xia et al. (2022)]. Specifically, the inter-sample similarity of the selected coreset is minimized in return for an overall high diversity. [Jiang et al. (2024c)] propose to preserve informative subset with the learning complexity (see Eq. 11) and implicitly puts constraints on its diversity via sampling on the k-means clusters:

$$D^{dist}(S) = \frac{1}{|S|}\sum_{x_i \in S} \min_{j \neq i} d(x_i, x_j) \geq C, \tag{43}$$

where C denotes the constant that controls the degree of diversity. A larger C represents the larger diversity of the dataset. The detailed procedure can be found in Alg. 4. It can be seen that datapoints are first sampled cluster-by-cluster to ensure a more balanced data distribution. Then, a percentile-based selection is performed on each cluster to select datapoints with easy learning complexity.

Algorithm 4 Easy and Diverse First Sampling [Jiang et al. (2024c)]

Require: data $x_i \in S$, existing pool S_b^0, a budget b, and the number of clusters K

1: Initialize $S_b = S_b^0$
2: $\arg\min_C \sum_{j=1}^{K} \sum_{x_i \in C_j \subset S} \|\frac{g(x_i)}{\|g(x_i)\|} - \mu_j\|^2$,
$\quad \mu_j = \frac{1}{|C_j|}\sum_{x_i \in C_j} \frac{g(x_i)}{\|g(x_i)\|}$.
3: **for** $j = 1$ to K **do**
4: $\quad S_b^j = \{x_i | \hat{F}_{\tilde{S}}(\tilde{S}(x_i)) \leq \frac{b}{K}, x_i \in C_j\}$
5: $\quad S_b = S_b \cup S_b^j$
6: **end for**
7: **return** S_b

[Bukharin & Zhao (2023)] propose the quality-diversity instruction tuning (QDIT). It also uses FL functions for diversity measure of the subset S_b:

$$D^{FL}(S_b) = \sum_{x_j \in S} \max_{x_i \in S_b} sim(g(x_i), g(x_j)), \tag{44}$$

where $sim(\cdot, \cdot)$ denotes the similarity function (e.g., cosine similarity). If the selected S_b can be well-representative of the entire set S, then S_b is assumed of high diversity. Given quality scores defined by Eq. 21, the detailed mechanism of QDIT is described in Alg. 5 with greedy approximation. Different from the previous sequential setup that respectively prioritizes quality and diversity in two successive steps, QDIT adopts a dynamic weighting over the GPT scoring-based quality and the FL-based diversity for selection.

[Liu et al. (2023b)] adopt the quality score-first and diversity-aware data selection method (DEITA), where all datapoints are first scored and sorted by quality measurement, and then selected by a geometry-based

Algorithm 5 QDIT sampling [Bukharin & Zhao (2023)]

Require: data $x_i \in S$, a budget b, and the trade-off hyper-parameter α

Initialize $S_b = \emptyset$
for $t = 1$ to b **do**
  $u = \arg\max_{x_i \in S \setminus S_b} (1-\alpha) \cdot D^{FL}(S_b \cup \{x_i\}) + \alpha \cdot GPTScore_i$
  $S_b = S_b \cup \{u\}$
end for
return S_b

heuristic criterion (i.e., Repr Filter). Specifically, it considers that for each chosen datapoint in S_b, its kNN_i^1 (Eq. 30) should be above a certain threshold τ so that the overall diversity $D^{kNN}(S_b)$ (Eq. 32) can be improved. As shown in Alg. 6, the quality and complexity of each sample x_i are respectively measured by the trained complexity scoring model θ_C and the quality scoring model θ_Q with prompts p_C and p_Q. Then, samples with high G_{CQ} are prioritized but only those dissimilar ones can be kept for the diversity of S_b.

Algorithm 6 DEITA Sampling [Liu et al. (2023b)]

Require: data $x_i \in S$ and a budget b

Compute the combined complexity and quality score $G_{CQ}(x_i) = G(x_i, p_C|\theta_C) \cdot G(x_i, p_Q|\theta_Q)$
$u = \arg\max_{x_i \in S} G_{CQ}(x_i|\theta)$
Initialize $S_b = \{u\}$
$S = S \setminus \{u\}$
while $|S_b| < b$ **do**
  $u = \arg\max_{x_i \in S} G_{CQ}(x_i|\theta)$
  if $d(g(u), g(N_0(u)) > \tau, N_0(u) \in S_b$ **then**
    $S_b = S_b \cup \{u\}$
  end if
  $S = S \setminus \{u\}$
end while
return S_b

Another series of geometry-based methods focus on the organization of data structures via developing clustering-based sampling techniques [Citovsky et al. (2021); Tirumala et al. (2024); Axiotis et al. (2024); Shao et al. (2024); Alcoforado et al. (2024); Saranathan et al.]. With respect to the clustering criterion, traditional methods employ topic modeling with LDA [Blei et al. (2003); Raghuveer et al. (2012); Bui et al. (2017)], NMF [Lee & Seung (2000); Wang & Zhang (2012); Shen & Si (2010); Lazar & Doncescu (2009)], TF-IDF [Sparck Jones (1972); Bafna et al. (2016); Patil & Atique (2013); Roul et al. (2014)], and latent concepts [Xie et al. (2021)] to assign text corpus into thematic clusters. Most recent studies exploit sentence encoding methods [Reimers & Gurevych (2019); Feng et al. (2020)] to perform clustering in the embedding space, where the vanilla k-means clustering and its variants [Sinaga & Yang (2020); Kanungo et al. (2000); Bandyapadhyay & Varadarajan (2015)], DBSCAN [Deng (2020); Khan et al. (2014); Creţulescu et al. (2019)], and spectral clustering [Bach & Jordan (2003); Von Luxburg (2007); Jia et al. (2014)] are widely used. Specifically, [Tirumala et al. (2024)] propose to use SemDeDup [Abbas et al. (2023)] to remove semantically similar examples for deduplication, which provides a basis of diversity sampling. Then, k-means clustering is performed in the embedding space. In each cluster, the prototype-based sampling technique [Sorscher et al. (2022)] is used. The "prototypical" samples, whose distance to their assigned cluster centers are small, should be discarded first to allow more "outliers" to be kept in S_b during iterative sampling (see Alg. 7). It is noted that two times of clustering operations are conducted, where the semantic deduplication and prototype-based sampling are respectively performed on K_1 and K_2 clusters.

[Axiotis et al. (2024)] propose a k-means cluster-based sensitivity sampling technique. For each datapoint in a cluster, both its distance to the cluster center and a proxy evaluation loss [Feldman & Langberg (2011)] measured on that cluster center contribute proportional to the probability of being chosen. [Shao et al. (2024)] propose the balanced ClusterClip sampling. It first performs k-means clustering and then sample datapoints

Algorithm 7 D4 Sampling [Liu et al. (2023b)]

Require: data $x_i \in S$, a budget b, the number of clusters for SemDeDup K_1 and the number of clusters for prototypicality K_2

Initialize $S_d = \emptyset, S_b = \emptyset$
$\arg\min_C \sum_{j=1}^{K_1} \sum_{x_i \in C_j \subset S} \| \frac{g(x_i)}{\|g(x_i)\|} - \mu_j \|^2$,
    $\mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} \frac{g(x_i)}{\|g(x_i)\|}$.
for $j = 1$ to K_1 **do**
    $C_j^v = \emptyset$
    while $|C_j^v| < |C_j|$ **do**
        $u = \arg\min_{x_i \in C_j \setminus C_j^v} sim(g(x_i), \mu_j)$
        if $\max_{x_i \in C_j} sim(g(u), g(x_i)) < \tau$ **then**
            $S_d = S_d \cup \{u\}$
        end if
        $C_j^v = C_j^v \cup \{u\}$
    end while
end for
$\arg\min_C \sum_{j=1}^{K_2} \sum_{x_i \in C_j \subset S_d} \| \frac{g(x_i)}{\|g(x_i)\|} - \mu_j \|^2$,
    $\mu_j = \frac{1}{|C_j|} \sum_{x_i \in C_j} \frac{g(x_i)}{\|g(x_i)\|}$.
for $j = 1$ to K_2 **do**
    $S_b^j = \{x_i | \hat{F}_d(d(x_i, \mu_j) > \frac{b}{K_2}, x_i \in C_j\}$
    $S_b = S_b \cup S_b^j$
end for
return S_b

uniformly from each cluster. Different from the uniform sampling, the proposed ClusterClip puts constraints on the maximum number of each cluster being sampled, and therefore avoids overfitting of small clusters.

[Alcoforado et al. (2024)] comprehensively compare different geometry-based diversity sampling techniques such as similarity or distance-based greedy sampling and clustering-based sampling. It proposes three approaches to select subsets S_b for human annotation: 1) reverse semantic search, 2) ordered clustering, and 3) limited lexical similarity. For the reverse semantic search, two datapoints (x_i, x_j) that share the least semantic similarity are first sampled as S_b^0 and then iterative selection of the next most dissimilar element from S is added into S_b^0. Its implementation is quite similar to the k-center greedy algorithm (see Alg. 2) except for the initialization of S_b^0. For the limited lexical similarity approach, the first sample x_0 is chosen randomly for initialization of S_b^0. For the remaining $b-1$ quota, each sample x_i is also randomly chosen from $S \backslash S_b$ as long as $sim(x_i, x_{i-1}) \leq \tau$, where $sim(\cdot, \cdot)$ here denotes the lexical similarity such as BLEU [Papineni et al. (2002)] and ROUGE scores [Lin (2004)]. The ordered clustering applies a hierarchical and density-based clustering algorithm like HDBSCAN [Campello et al. (2013)] on all samples and sequentially (i.e., from large to small clusters) choose the samples of the lowest membership in each cluster into the subset S_b. Experimental results show that the reverse semantic search performs most consistently and competitively, while the limited lexical similarity is sensitive to the hyper-parameter threshold τ. The ordered clustering is not robust across datasets and fails to select high-quality samples.

Remark Geometry-based sampling is intuitive and effective in diversity control. Most solutions to optimizing the overall diversity can be reformulated as variants of an iterative, similarity or distance-based, greedy sampling technique. Clustering does play an explanatory role in deciphering the embedding structures, making it easier and preciser to control the proportion of selection.

4.4 Bilevel Optimization-based Coreset Sampling

Overview The selection of coreset can also be viewede as a bilevel optimization problem [Colson et al. (2007); Zhang (2024); Sinha et al. (2017); Borsos et al. (2020); Killamsetty et al. (2021b;c); Zhang et al. (2022);

Borsos et al. (2024); Pan et al. (2024)] that consists of two loops: 1) the outer loop of optimizing the hard masks or soft weights for selecting the subset S_b from S; 2) the inner loop of optimizing the model parameters θ on S_b. Without lose of generalizability, the bilevel optimization with the self-supervised language modeling loss can be written as follows:

$$\begin{aligned} S_b &= \arg\min_{S_b^{'} \subset S} \sum_{x_i \in S_b^{'}, \theta=\theta^*} NLL_i^{A|Q}, \\ \text{s.t. } \theta^* &= \arg\min_{\theta} \sum_{x_i \in S_b^{'}} NLL_i^{A|Q}. \end{aligned} \tag{45}$$

Technical Details The retrieve method proposed by [Killamsetty et al. (2021c)] takes both labeled and unlabeled datasets into consideration, where the self-supervised loss from the unlabeled set (e.g., consistency regularization [Xie et al. (2020); Wang et al. (2021c)] and entropy regularization [Zhao et al. (2020b); Grandvalet & Bengio (2004); Erkan & Altun (2010)]) contributes to the inter-level and outer-level optimization as well. To improve the robustness, Glister [Killamsetty et al. (2021b)] optimizes the outer-level coreset selection on the additionally prepared validation set for the minimized validation loss. [Li et al. (2023e)] further emphasize the role of the validation set in bilevel optimization. It not only computes the loss on the validation set for adversarial training, but also introduces gradient matching [Killamsetty et al. (2021a)] where the gradient of the model on the selected subset S_b should be close to that on the entire S.

[Borsos et al. (2024)] reformulate the coreset sampling as a cardinality-constrained bilevel optimization problem. It proposes greedy forward selection and first-order methods that apply to any twice differentiable models. Variants of the solution for acceleration are extended: 1) binary weights, inverse-hessian-vector product approximations, and batch-wise selection; 2) small proxy models for fast estimation; 3) enforced sparsity-inducing penalty in the outer loop.

The ScaleBiO [Pan et al. (2024)] specifically address the data reweighting problem for large-scale LLM instruction tuning. It also prepares an extra validation set S^{val} for the minimization of the outer loop. ScaleBio transforms the bilevel optimization into the single loop framework with an outer-level problem plus a constraint of the inner-level problem. A multiplier $\alpha > 0$ and a proxy u for optimizing the original inner loop (i.e., model weights θ) are introduced into the minimax formulation [Kwon et al. (2023); Lu & Mei (2024)].

In contrast to a fixed budget b, [Xia et al. (2024b)] propose a lexicographic bilevel-optimization method [Borsos et al. (2020); Killamsetty et al. (2021b;c)] where the inner loop optimizes model parameters and the outer loop optimizes data selection. During optimization of the data selection mask, the loss terms are relaxed to allow the size of the final coreset smaller than b.

Remark The bilevel optimization methods often involve regularization tricks as a relaxation to the original problem with nested outer-inner loops. Compared with the boolean selection, soft weights-based objective guarantees a higher level of diversity as each sample contributes more or less to the overall optimization.

5 Importance-based Selection

This section provides the review of methods on importance measurement and selection. By importance we mean the necessity of adding one instruction-response sample into the training set. Due to the pre-training nature of LLMs, a wide range of materials have been "parameterized" as internal knowledge and therefore several common tasks can be correctly solved without additional fine-tuning. In this case, alignment is not required for easy samples but becomes indispensable for difficult ones. The selected datapoints provide supplementary knowledge to activate the pre-trained LLMs on following complex instructions. Generally, the evaluation function $q(x_i)$ in the context of importance can be alternatively implemented as: 1) the complexity of the datapoint itself q_C that can be perceived by the task definition and the model's uncertainty, 2) the contribution of each datapoint to the overall performance q_P that can be reflected in the losses and errors during learning, and 3) the degree of gradient matching q_G that indicates influential datapoints for effective selection. Consequently, we can unify the importance-based measurements as:

$$q(x_i) = f_i(q_C(x_i), q_P(x_i), q_G(x_i)), \tag{46}$$

where f_i denotes the aggregation function. It is noted that most existing methods simply choose one of the above evaluation implementations and therefore f_i can be designed as an exclusive choice function.

5.1 Hand-crafted Indicators

Overview Existing researches on importance measurement of datapoints often stem from two aspects: 1) from the perspective of a datapoint itself, i.e., the difficulty or complexity of each datapoint and the amount of information it provides; 2) from the perspective of the model under development, i.e., the necessity of learning from such a datapoint based on the current performance and confidence (uncertainty), Most hand-crafted indicators are proposed to analyze the text difficulty.

Technical Details The readability indices [Young & Shishido (2023)] can be used to assess both quality (see Sec. §3.1) and difficulty of text samples. Specifically, samples with intricate grammar, advanced vocabulary, and inference dependency are deemed as difficult ones and can be used to evaluate robustness of models across benchmarks of various difficulty levels [Smith & Johnson (2020); Kiela et al. (2021); Ethayarajh et al. (2022); Belinkov & Glass (2019); Nie et al. (2019); Ribeiro et al. (2020)]. For specialized domains such as solving maths problems, the education level (e.g., elementary-level, high school-level, and university-level) determines the difficulty of samples [Patel et al. (2021); Huang et al. (2016); Koncel-Kedziorski et al. (2016)].

One of the pioneering studies on readability scores for difficulty assessment is to compute the percentage of difficult or easy words in one sentence [Klare (1974); Begeny & Greene (2014)]. The words on a pre-defined list are counted as familiar words, and those not listed are unfamiliar, advanced words. Besides, the average number of syllables per word, the number of single-syllable words, and the number of multi-syllable words are also indicative in assessing the text materials [Connatser (1999); Carrell (1987); Zakaluk & Samuels (1988); Dale & Chall (1949)]. Notably, there exist three representative readability metrics: 1) the Dale Chall formula [Chall & Dale (1995)], 2) the flesch reading ease [Flesch (1948)], and 3) the gunning fog index [Gunning (1952)]. Given these metrics, [Saranathan et al.] conduct a thorough analysis on existing NLP datasets S to select the most challenging subsets for efficient evaluation of LLMs. The easiest and hardest samples from the TruthfulQA [Lin et al. (2021)] via these indicators are confirmed positively correlated with the actual complexity. The selection of difficult instruction-response pairs via Eq. 8 allows a wider distribution of performance for the models under investigation. The introduction of difficult datapoints into the subset S_b helps keep the relative rank of different models unchanged compared with that measured on the entire set S.

Remark The difficulty indices help comprehensively analyze the robustness of models across samples and datasets. In addition, it also presents guidelines of curating and constructing discriminating NLP benchmarks.

5.2 Model-based Indicators

Overview To avoid potential confusion, the model-based importance indicators discussed in this section are mainly categorized as three kinds: 1) uncertainty-based; 2) reward score-based; and 3) data model-based. Methods that employ training/inference losses, errors (metrics), and gradients are not included despite their involvement of the language model for importance sampling.

Technical Details Inspired from uncertainty indicators, [Siddhant & Lipton (2018); Kung et al. (2023); Nieth et al. (2024)] propose the prompt uncertainty which measures the disagreement between model responses on different perturbed versions of the same instruction:

$$U_i^{\text{prompt}} = -\frac{1}{K}\sum_{k=1}^{K}\sum_{j=t}^{|x_i|}|P(x_{i(j)}|x_{i(<j)};\theta) - P(x_{i(j)}|\tilde{x}_{i(<j)}^k;\theta)|, \tag{47}$$

where K denotes the number of perturbations and $\tilde{x}_i^k$ is the k-th perturbed prompt. Note that only the instruction part $x_{i(<t)}$ is perturbed and sent to the model for the following likelihood measurement on the

original response $x_{i(j)}, j = t, t+1, ..., |x_i|$. Samples with high prompt uncertainty should be chosen for fine-tuning since the model does not perform consistently on such instructions.

[Jiang et al. (2023b)] target at the over-confidence problem of LLMs after instruction tuning [Kadavath et al. (2022)], and propose the CAPE to calibrate the uncertainty with augmented prompt ensembles. They first transform all discriminative and generative tasks into multiple-choice problems, and use the LLM's predicted probabilities over answer choices (e.g., A, B, C) for uncertainty estimation. Then, prompt augmentations are performed via paraphrasing of the template, permutation of the in-context examples and the answer choices. Multiple predictions over the answer choices are collected with the augmented prompts as inputs, which helps calibrate the uncertainty in an ensemble manner. Such calibrated uncertainty tells if an instruction-tuned LLM simply memorizes the response to a given prompt rather than truly understanding the instruction. Therefore, it can be used to precisely choose important datapoints with high uncertainty.

Apart from the uncertainty, the reward model can also be used beyond quality scorer. Since most of the knowledge and capabilities are acquired during pre-training [Zhou et al. (2024a)], the instruction tuning datasets are aimed at aligning the behavior of models with human preference and expectations. Therefore, for any given instruction x_i, if the generated response is of high quality, then the necessity of fine-tuning on this instruction is low. Accordingly, x_i is deemed as "unimportant" and will not be chosen into the subset. In that case, the language model parameterized as θ is first prompted with $x_{i(<t)}$ to generate the response $\hat{x}^{\theta}_{i(\geq t)}$. Then, a reward model parameterized as ϕ acts as a necessity evaluation model:

$$\hat{R}_i = r_\phi(x_{i(<t)}, \hat{x}^{\theta}_{i(\geq t)}), \tag{48}$$

Samples whose necessity score $\hat{R}_i$ below a pre-determined threshold are selected via Eq. 7, implying that the model θ does not own the capabilities to handle x_i and requires fine-tuning.

Another series of model-based importance estimation methods are based on datamodels [Ilyas et al. (2022); Park et al. (2023); Jain et al. (2023); Kang et al. (2024); Chhabra et al. (2024); Saunshi et al. (2022); Ye et al. (2024)], where the contribution of each datapoint to the model's behavior is estimated. The datamodels can be implemented in any machine learning model which targets at predicting the influence of each datapoint on the performance of the trained model [Koh & Liang (2017); Jain et al. (2022); Liu et al. (2024b); Picard et al. (2024); Bae et al. (2024); Covert et al. (2024)].

[Engstrom et al. (2024)] propose to use datamodels to select subsets that maximize the overall performance. Specifically, it chooses the subset $S_b \subset S, S = \{x_1, x_2, ..., x_{|S|}\}$ by estimating the loss of the model trained on it. Out of simplicity, the datamodel τ_{θ_x} can be implemented as a linear model and it learns to approximate the actual loss via the TARK estimator [Park et al. (2023)]:

$$\begin{aligned}
&\theta_{x_j} = \arg\min_{\theta} \hat{\mathbb{E}}^{(m)}_{S_i \sim S_b \subset S}[L_{reg}(\tau_\theta(\mathbb{1}_{S_i})), \mathcal{L}_{x_j}(S_i)], \\
&\mathbb{1}_{S_b} \in \{0,1\}^{|S|}, \quad (\mathbb{1}_{S_b})_i = \begin{cases} 1, & \text{if } x_i \in S_b, \\ 0, & \text{otherwise.} \end{cases}, \\
&\tau_{\theta_x}(\mathbb{1}_{S_b}) = \theta_x^T \mathbb{1}_{S_b},
\end{aligned} \tag{49}$$

where $\mathcal{L}_{x_j}(S_i)$ denotes the loss of the model (trained on S_i) on the sample x_j. The $\hat{\mathbb{E}}^{(m)}$ is a m-sample empirical expectation and $L_{reg}(\cdot,\cdot)$ is a regression loss function (e.g., mean squared error). Intuitively, what the datamodel τ_{θ_x} does is to approximate the real loss $\mathcal{L}_{x_j}(S_i)$ under various compositions of subsets $S_i \sim S_b$. Given any subset $S_b^{'}$, the averaged loss approximated by the datamodel on all $x_j \in S_{eval}$ is calculated on the

evaluation set S_{eval} and minimized to find the optimal $S_b, |S_b| = b$:

$$
\begin{aligned}
S_b &= \arg\min_{S_b^{'} \subset S} \hat{\mathcal{L}}_{S_{eval}}(S_b^{'}), \\
\hat{\mathcal{L}}_{S_{eval}}(S_b^{'}) &= \hat{\mathbb{E}}^{(n)}_{x_j \sim S_{eval}}[\tau_{\theta_{x_j}}(\mathbb{1}_{S_b^{'}})] \\
&= \frac{1}{|S_{eval}|} \sum_{x_j \in S_{eval}} \theta_{x_j}^T \mathbb{1}_{S_b^*} \\
&= \mathbb{1}_{S_b^*}^T (\frac{1}{|S_{eval}|} \sum_{x_j \in S_{eval}} \theta_{x_j}).
\end{aligned} \tag{50}
$$

The importance of $x_i \in S$ is therefore measured by $\frac{1}{|S_{eval}|} \sum_{x_j \in S_{eval}} \theta_{x_j}$ and its smallest b elements are chosen for the minimum loss $\hat{\mathcal{L}}_{S_{eval}}$.

[Liu et al. (2024b)] also propose a simulation-based [Guu et al. (2023)] linear datamodel that correlates the training samples with the validation or test set loss. A featured simulator, namely GPTfluence, models the training dynamics (e.g., loss, BLEU and ROUGE scores) across time via an n-th order Markov process. It extracts representations $g(x_i), x_i \in S$ from BERT or GPT, and generates both multiplicative and additive factors to reflect the influence of any training example on the testing set. The testing performance ϕ_t at any time t is affected by: 1) its performance at preceding n times and 2) the current training batch c_t:

$$
\begin{aligned}
\phi_t(x_k) &= \sum_{j=1}^{n} \alpha_j(c_t)\phi_{t-j}(x_k) + \beta(c_t), \forall x_k \in S_{eval}, \\
\alpha_j(c_t) &= \sum_{i=1}^{|c_t|} A_{i,j}, \quad \beta(c_t) = \sum_{i=1}^{|c_t|} B_i, \forall x_i \in c_t \subset S, \\
A_{ij} &= \langle \mathbf{W}_{(j)}^T g(x_i)_j, \mathbf{U}_{(j)}^T g(x_k) \rangle_F, \\
B_i &= \langle \mathbf{W}'^T g(x_i)_j, \mathbf{U}' g(x_k) \rangle_F,
\end{aligned} \tag{51}
$$

where $\mathbf{W}_{(j)}^T, \mathbf{U}_{(j)}^T, \mathbf{W}', \mathbf{U}'$ are learnable weights which are optimized by minimizing $\sum_{t=1}^{T}(y_t - \phi_t(x_k))^2$ with y_t being the ground-truth metric score monitored during training at step t. The $\langle \cdot, \cdot \rangle_F$ denotes the Frobenius inner product. Samples that reduce the evaluation loss the most are selected as influential data.

Instead of performing off-line selection, [Yu et al. (2024)] propose MATES where a small datamodel continuously selects the most effective subset for the current training of the LLM. The datamodel, like a partner, is updated alternatively to adapt to the constantly changing preferences of the model under development.

Unlike previous datamodels that predict the influence of datapoints via the testing performance of the model, [Xie et al. (2023)] propose the DSIR with importance scores estimated by the distributional resemblance. It simply assumes that training samples that resemble the evaluation set are important, and these datapoints should be selected with higher probability. Given the hashed n-grams features $h(x_i) \in \mathbb{N}^m$ of x_i, its importance

score w_i is calculated as:

$$
\begin{aligned}
w_i &= \frac{\hat{w}_i}{\sum_{i=1}^{|S|} \hat{w}_i}, \\
\hat{w}_i &= \frac{\hat{p}_{feat}(h(x_i))}{\hat{q}_{feat}(h(x_i))}, \\
\hat{p}_{feat}(h(x_i)) &= \prod_{j=1}^{m} \gamma_j^{h(x_i)_j}, \\
\hat{q}_{feat}(h(x_i)) &= \prod_{j=1}^{m} \beta_j^{h(x_i)_j}, \\
\hat{\gamma} &= \frac{1}{\sum_{x_i \in S_{eval}} \mathbf{1}^T h(x_i)} \sum_{x_j \in S_{eval}} h(x_j), \\
\hat{\beta} &= \frac{1}{\sum_{x_i \in S} \mathbf{1}^T h(x_i)} \sum_{x_j \in S} h(x_j),
\end{aligned}
\tag{52}
$$

where S and S_{eval} respectively denote the training set and the evaluation set. Given the budget b, the subset S_b is obtained by importance-weighted sampling without replacement b times. [Zhang et al. (2023c)] also proposes to use a independent-cascade diffusion model [Li et al. (2018); Du et al. (2014)] to mimic the information diffusion process upon a directed graph on embeddings of datapoints. The most influential datapoint are selected for annotation and serve as in-context learning examples for LLMs.

Remark Compared with uncertainty and reward score, datamodel-based importance indicators are more correlated with the downstream performance since the task-specific evaluation set is introduced to provide feedback on the selection scheme.

5.3 Loss and Error-based Coreset Sampling

Overview During training, samples that contribute more to the total loss or cause worse performance are considered more important. In the light of this statement, the influence of each datapoint can also be measured via the losses and errors for coreset sampling. Compared with the datamodel-based measurement that estimates individual importance with a specifically designed datamodel, loss and error-based measurement is performed with the same LLM under development.

Technical Details One kind of methods that record the errors of each sample during training to estimate importance is forgetting score or forgetting event [Toneva et al. (2018)]. It counts how many times the forgetting happens with the iteration of training step t. For any given sample x_i in a batch B ($x_i \in B \subset S$), if the previous accuracy acc_i^{t-1} surpasses the current accuracy acc_i^t ($acc_i^t > acc_i^{t+1}$), then the example x_i undergoes a forgetting event. Conversely, a learning event occurs if $acc_i^t < acc_i^{t+1}$. The number of forgetting events implies whether the sample is difficult and indispensable for training. An example x_i is defined as unforgettable if it satisfies:

$$
Unforget_i = \begin{cases} 1, & \exists t^* < \infty, \text{ s.t. } acc_i^t < acc_i^{t+1} \\ & \text{and } \forall k \geq t^*, acc_i^k > acc_i^{k-1}, \\ 0, & \text{otherwise.} \end{cases}
\tag{53}
$$

The easy samples with $Unforget_i = 1$ can be simply discarded and the important subset $S_b = \{x_i | Unforget_i = 0, x_i \in S\}$ is selected for training. Recent studies on both pre-training and instruction tuning have investigated the effectiveness of using the forgetting score for efficient data pruning [Sorscher et al. (2022); Paul et al. (2021); Zhang et al. (2023a); Jin & Ren (2024a); Maini et al. (2022)].

In contrast to the term "forgetting", researchers introduce the concept "memorization" [Feldman (2020); Tirumala et al. (2022); Antoniades et al. (2024)] for analysis on the generalization of deep models [Zhang et al. (2021)]. The memorization of training samples is necessary for reducing close-to-optimal generalization

error especially when a long-tailed disttribution is observed for the training set [Feldman (2020)]. Specifically, the amount of label memorization on the instruction-response pair $(x_{i(<t)}, x_{i(\geq t)})$ is defined as follows:

$$Memo_i = \frac{1}{|x_i| - t} \sum_{j=t}^{|x_i|} (P(x_{i(j)}|x_{i(<j)}; \theta^S) - P(x_{i(j)}|x_{i(<j)}; \theta^{S \setminus x_i})), \tag{54}$$

where θ^S and $\theta^{S \setminus x_i}$ respectively refer to the language model parameters optimized with the entire set with and without x_i. Accordingly, the influence of the sample x_i on other samples $x_k, x_k \neq x_i$ can be defined as [Feldman & Zhang (2020)]:

$$Infl_{ik} = \frac{1}{|x_k| - t} \sum_{j=t}^{|x_k|} (P(x_{k(j)}|x_{k(<j)}; \theta^S) - P(x_{k(j)}|x_{k(<j)}; \theta^{S \setminus x_i})), \tag{55}$$

where $x_{k(<t)}$ and $x_{k(\geq t)})$ respectively denote the instruction and response part of x_k. In practice, the memorization and influence scores are approximated via batch-wise sampling where N batches $B_1, B_2, ..., B_N$ are sampled from S with $|B_i| = n$. For each batch B_i, a language model parameterized as θ^{B_i} is trained to compute the memorization and influence scores of each sample x_i. It is noted that some batches contain x_i and the others do not. Therefore, the two probability terms in Eqs. 54 and 55 are respectively averaged over multiple probability outputs of the models trained on batches with and without x_i. [Sorscher et al. (2022)] confirm that memorization scores (Eq. 54) demonstrate stronger performance on pruning the dataset into a significantly smaller subset S_b than random sampling, EL2N (Eq. 13), and influence scores (Eq. 55). [Suzuki et al. (2023)] and [Schoch et al. (2023)] also follow [Feldman & Zhang (2020)] to select the high-quality influential subset for LLM training.

Furthermore, [Chen et al. (2024b)] use the evaluation loss to check whether the current task requires certain skills or capabilities that can be obtained by learning from the prerequisite tasks. For each task, it selects the skill-dependent datapoints that reduce evaluation loss. [Mishra & Sachdeva (2020)] propose a rather simple method that adopts a proxy model (e.g., logistic regression and SVM) to train on the randomly selected subset S_b and evaluate on the remaining set $S \setminus S_b$. Such process iterates over multiple times to ensure that each sample is at least validated once. The probability of each sample being correctly predicted is used as importance measurement. Likewise, [Lin et al. (2022)] also quantify the average marginal effect (AME) as influence of x_i. It can be viewed as a variant of shapley value [Jia et al. (2019); Ghorbani & Zou (2019); Schoch et al. (2023); Kwon & Zou (2021)]. Different subsets are randomly sampled to train multiple submodels and each submodel is evaluated for jointly estimating the AME via LASSO regression [Lecué & Mendelson (2018)].

Remark The loss and error-based selection methods are intuitive and effective to select the datapoints with high difficulty and influence. To accelerate the computation of marginal effect (gain) of each datapoint, iterative approximations can be adopted with small proxy models.

5.4 Gradient-based Coreset Sampling

Overview Since gradients directly affect the optimization of language models, two kinds of intuitive methods for data selection are presented: 1) gradient matching [Zhao et al. (2020a); Killamsetty et al. (2021a); Jiang et al. (2023d); Zhao & Bilen (2023); Du et al. (2024); Balles et al. (2022); Zhang et al. (2024a)], i.e., the gradients of the entire set S being approximated by the weighted gradients of the subset S_b, and 2) gradient-based influence [Pruthi et al. (2020); Brophy et al. (2023); Koh & Liang (2017); Basu et al. (2020); Picard et al. (2024); Alaa & Van Der Schaar (2020)], i.e., the influence of each sample x_i on a testing datapoint x_t being measured by upweighted gradient multiplication. Specifically, the gradient matching aims

to minimize the difference below:

$$\theta^*, S_b^* = \arg\min_{\theta,S} \; d(\frac{1}{|S|}\sum_{x_i \in S} \nabla_\theta NLL_i^{A|Q}, \\ \frac{1}{\sum_i w_i}\sum_{x_i \in S_b} w_i \nabla_\theta NLL_i^{A|Q}), \; S_b \in S, w_i > 0, \tag{56}$$

where $d(\cdot,\cdot)$ denotes the distance measurement and w_i is the weight for the gradient of x_i.

The gradient-based influence methods, on the other hand, aim at selecting the most influential datapoints in terms of the variation of model parameters θ. Given the optimal parameters θ^*, the updated parameters $\theta^\epsilon_{\{x_i\}}$ by up-weighting the loss of x_i with ϵ can be derived as the first-order Taylor series expansion as follows:

$$\theta^\epsilon_{x_i} = \arg\min_\theta \frac{1}{|S|}\sum_{x_j \in S} NLL_j^{A|Q} + \epsilon NLL_i^{A|Q}, \\ \theta^\epsilon_{x_i} \approx \theta^* - \epsilon H_{\theta^*}^{-1} \nabla_\theta NLL_i^{A|Q}, \tag{57}$$

where H_{θ^*} represents the Hessian with respect to the model parameters θ^*. Accordingly, the influence function of a sample x_i on the model parameters and its effect on the performance of a particular sample x_j can be respectively denoted as:

$$InflF_i = \frac{\mathrm{d}\theta^\epsilon_{x_i}}{\mathrm{d}\epsilon}|_{\epsilon=0} = -H_{\theta^*}^{-1}\nabla_\theta NLL_i^{A|Q}, \\ InflF_{ij} = -\nabla_\theta NLL_j^{A|Q^T} H_{\theta^*}^{-1} \nabla_\theta NLL_i^{A|Q}. \tag{58}$$

The importance indicator $InflF_{ij}$ approximately measures the change of the loss on x_j when x_i is removed from the training set. To expedite the computation of Hessian matrix for large models, a combination of Hessian-vector product and optimization techniques are developed [Pearlmutter (1994); Nilsen et al. (2019); Mathieu & LeCun (2014); Agarwal et al. (2016); Shewchuk et al. (1994)].

Another kind of influence score is defined as the expected gradient norm (GraNd score) [Paul et al. (2021); Kirsch (2023); Böther et al. (2023)], where the GraNd score controls the contribution of a training sample to the change of the loss.

$$GraNd_i = \mathbb{E}_\theta \|\nabla_\theta NLL_i^{A|Q}\|_2 \tag{59}$$

Experiments [Paul et al. (2021)] suggest that the GraNd score (Eq. 59) can be well approximated by EL2N score (Eq. 13) for efficient data pruning.

Technical Details [Xia et al. (2024a)] propose to find the most influential training data that resemble the testing set the most via low-rank gradient similarity search. [Tan et al. (2024a)] introduce the moving-one-sample-out (MoSo) by pinpointing the least informative samples via gradient-based influence assessment. To avoid the costly retraining procedure by iteratively moving one sample out, a gradient-based approximator is proposed to select samples whose gradients are consistently aligned with the average gradients of the entire training set.

For the detailed definition of distance measure of Eq. 56, [Everaert & Potts (2023)] exploit the KL-divergence to measure the difference between the selected subset and the testing set. Note that here the objective is to approach the distribution of the testing set rather than the entire training set. [Killamsetty et al. (2021a)] speed up the gradient matching between the selected dataset and the validation set via an orthogonal matching pursuit algorithm. [Lin et al. (2024)] apply gradient-based influence scores on recommendation datasets for effective LLM instruction tuning. [Schioppa et al. (2021)] choose a different way [Arnoldi (1951)] to accelerate the computation of the inverse Hessian matrix in Eq. 57 and successfully scales up the influence scoring for LLMs with several hundreds of millions of parameters. [Grosse et al. (2023)] use the influence functions to study the generalization properties of LLMs To scale up influence functions for LLMs up to 52 billions, an approximation technique via Eigenvalue-corrected Kronecker-Factored Approximate Curvature (EK-FAC) [George et al. (2021)] to efficiently find the most influential samples to the pre-trained LLMs over

Table 1: Statistics of datasets in existing representative data assessment and selection methods.

Methods	Quality	Diversity	Importance	Training Set	Training Set Size
IFD [Li et al. (2023a)]	✓	✗	✗	Alpaca WizardLM	52K 70K
LIFT [Xu et al. (2023b)]	✓	✓	✗	Open-Platypus CodeAlpaca	25K 20K
DQ [Zhou et al. (2023)]	✗	✓	✗	Alpaca	52K
PPL [Ankner et al. (2024)]	✓	✗	✗	The Pile Dolma	NA NA
InstructionMining [Cao et al. (2023)]	✓	✗	✗	OpenOrca Dolly	50K 15K
FL [Bhatt et al. (2024)]	✓	✓	✗	FLAN v2	99K
Alpagasus [Chen et al. (2023b)]	✓	✗	✗	Alpaca	52K
BSDetector [Chen & Mueller (2024)]	✓	✗	✗	SQuAD-N Emails-N DROP-N	NA NA NA
DEITA [Liu et al. (2023b)]	✓	✓	✗	Mixed(ShareGPT+UltraChat+WizardLM)	206K
AutoDS [Zhang et al. (2024c)]	✓	✗	✗	OpenWebMath	NA
Qurator [Wettig et al. (2024)]	✓	✗	✗	QuRatedPajama	260B tkn
ClusterClip [Shao et al. (2024)]	✗	✓	✗	OpenOrca Proof-Pile-2	4.2M 2.7M
QDIT [Bukharin & Zhao (2023)]	✓	✓	✗	UltraChat LMSYS Alpaca Mixed (Alpaca+OIG+Dolly) Dolly	1.3M 1M 52K 270K 15K
DsDm [Engstrom et al. (2024)]	✗	✗	✓	C4	NA
MATES [Yu et al. (2024)]	✗	✗	✓	C4	NA
DSIR [Xie et al. (2023)]	✗	✗	✓	The Pile	1.6B
Skill-it [Chen et al. (2024b)]	✗	✗	✓	RedPajama	1.2T tokens
LESS [Xia et al. (2024a)]	✗	✗	✓	Mixed(FLAN v2+Dolly+OpenAssistant+COT)	270K

maths and programming abilities, cross-lingual generalization, and role-playing behavior. [Zhao et al. (2021)] condense the datasets into small informative synthetic samples where the gradients of the model on the synthetic data are matching those on the real data of the entire training set.

Remark The gradient-based coreset sampling techniques are highly dependent on the LLMs under development, where the gradients describe the model's inherent knowledge and uncertainty about each datapoint. Despite the precision of gradient-based selection methods, it is noted that approximation is unavoidable for application on LLMs. The efficiency and accuracy of various approximation techniques should be considered.

6 Results and Discussions

In this section, we classify different methods according to their respective emphases, and then demonstrate their effectiveness in the selection of "high-standard" datapoints for instruction tuning. First, we provide explanations on the classification and selection of recent instruction tuning methods. Then, we summarize the representative methods with their data statistics in Tab. 1. Finally, we provide the detailed experimental results of each method respectively under our structures of quality, diversity, and importance.

Criteria on Classification and Selection of Existing Methods For the classification of existing methods into quality-based, diversity-based, and importance-based categories, we adhere to the following evidence: 1) the data characteristics emphasized in the motivation of their method development, and 2) the optimization objectives organized in this survey that are closest to those in their pipelines. For instance, the concept of diversity is stressed in DEITA [Liu et al. (2023b)] where a similarity-based filtering step is set to remove highly duplicated datapoints. Therefore, it falls under the category of diversity-based selection methods. DSIR [Xie et al. (2023)], on the other hand, directly estimates the importance of datapoints by matching the distribution between the selected subset and the target evaluation set. It is noted that many existing methods design compound, multi-facet selection criteria that strike a balance between quality, diversity, and importance. In this case, we choose the most representative methods under each category to

Table 2: Experimental results of quality-based selection methods are directly cited from their papers. BT denotes billions of tokens. WK, CR, LU, SPS, and RC respectively stand for compound datasets of World Knowledge, Commonsense Reasoning, Language Understanding, Symbolic Problem Solving, and Reading Comprehension.

Methods	Training Set	Model	Ratio/Size	Reported Results on Testing Sets				
				ARC	HellaSwag	MMLU	TruthfulQA	AlpacaEval
IFD [Li et al. (2023a)]	Alpaca	LLaMA 7B	Full	0.427	0.769	0.417	0.396	0.265
			5%	0.539	0.795	0.365	0.383	0.347
	WizardLM		Full	0.531	0.774	0.378	0.429	0.620
			10%	0.529	0.790	0.331	0.414	0.614
	Alpaca	LLaMA2 7B	Full	0.544	0.787	0.470	0.410	0.278
			5%	0.558	0.579	0.804	0.442	0.368
			10%	0.580	0.804	0.466	0.402	NA
			15%	0.564	0.574	0.807	0.464	NA
	WizardLM		Full	0.576	0.820	0.541	0.415	0.350
			5%	0.624	0.840	0.557	0.428	0.468
			10%	0.630	0.839	0.553	0.419	NA
			15%	0.624	0.835	0.556	0.434	NA
LIFT [Xu et al. (2023b)]				ARC	HellaSwag	MMLU	TruthfulQA	
	Open-Platypus	Mistral 7B	Random 15K	0.607	0.820	0.625	0.438	
			LIFT 15K	0.643	0.844	0.645	0.490	
				HumanEval	MBPP			
	Code-Alpaca	StarCoder 15B	Random 10K	0.381	0.431			
			LIFT 10K	0.550	0.495			
PPL [Ankner et al. (2024)]				WK	CR	LU	SPS	RC
	The Pile	MPT 1B	Full	0.155	0.103	0.281	0.035	0.112
			Low 50%	0.111	0.058	0.187	0.035	0.087
			Mid 50%	0.161	0.090	0.281	0.034	0.109
			High 50%	0.182	0.128	0.332	0.034	0.106
	Dolma		Full	0.165	0.123	0.289	0.036	0.080
			Low 50%	0.161	0.101	0.273	0.345	0.079
			Mid 50%	0.180	0.130	0.319	0.034	0.104
			High 50%	0.167	0.131	0.311	0.032	0.086
InstructionMining [Cao et al. (2023)]				ARC	HellaSwag	MMLU	TruthfulQA	
	OpenOrca & Dolly	LLaMA2 7B	IM 10K	0.567	0.798	0.499	0.483	
			IM 40K	0.544	0.801	0.526	0.498	
			Random 10K	0.548	0.796	0.490	0.516	
			Random 40K	0.548	0.799	0.512	0.500	
FL [Bhatt et al. (2024)]				MMLU	BBH			
	FLAN v2	LLaMA2 7B	Random 20K	0.443	0.390			
			FL 20K	0.451	0.383			
			Random 30K	0.449	0.394			
			FL 30K	0.471	0.411			
			Random 45K	0.460	0.394			
			FL 45K	0.476	0.413			
Alpagasus [Chen et al. (2023b)]				BBH	DROP	HumanEval	MMLU	
	Alpaca	LLaMA2 7B	Random 9K	0.319	0.259	0.116	0.369	
			Full 52K	0.330	0.259	0.117	0.409	
			Alpagasus 9K	0.338	0.260	0.122	0.388	
		LLaMA2 13B	Random 9K	0.386	0.334	0.152	0.450	
			Full 52k	0.387	0.338	0.157	0.479	
			Alpagasus 9K	0.389	0.344	0.159	0.461	
BSDetector [Chen & Mueller (2024)]				SQuQA-N	Emails-N	DROP-N		
	SQuAD-N	LLaMA2 7B Chat	Full	0.499	NA	NA		
			Auto-filter	0.599	NA	NA		
			Auto-correct	0.714	NA	NA		
	Emails-N		Full	NA	0.507	NA		
			Auto-filter	NA	0.497	NA		
			Auto-correct	NA	0.523	NA		
	DROP-N		Full	NA	NA	0.447		
			Auto-filter	NA	NA	0.474		
			Auto-correct	NA	NA	0.505		
AutoDS [Zhang et al. (2024c)]				MATH	GSM8K	BBH	ARC-E	ARC-C
	Open-WebMath	Mistral 7B	Random 2.5BT	0.143	0.441	0.565	0.842	0.567
			AutoDS 2.5BT	0.161	0.454	0.586	0.842	0.552
				LogiQ	BoolQ	NQ	MMLU	HellaSwag
			Random 2.5BT	0.310	0.838	0.292	0.522	0.622
			AutoDS 2.5BT	0.310	0.831	0.291	0.523	0.627
				PIQA	Winogrande	SciQ		
			Random 2.5BT	0.822	0.802	0.972		
			AutoDS 2.5BT	0.822	0.800	0.968		
QuRator [Wettig et al. (2024)]				RC	CR	WK		
	QuRated-Pajama	Sheared LLaMA 1.3B	Random 30BT	0.509	0.55	0.149		
			Qurator 30BT	0.521	0.555	0.152		

Table 3: Experimental results of diversity-based selection methods are directly cited from their papers.

Methods	Training Set	Model	Ratio/Size	Reported Results on Testing Sets			
				ARC	HellaSwag	MMLU	TruthfulQA
DEITA [Liu et al. (2023b)]	Mixed	LLaMA-13B	Random 10K	0.558	0.800	0.474	0.574
			DEITA 10K	0.595	0.820	0.606	0.550
		LLaMA2-13B	Random 10K	0.615	0.837	0.552	0.448
			DEITA 10K	0.589	0.821	0.553	0.546
		Mistral-7B	Random 10K	0.554	0.792	0.587	0.536
			DEITA 6K	0.578	0.803	0.619	0.598
				SuperGLUE	GSM8k	OBQA	MT-Bench
ClusterClip [Shao et al. (2024)]	OpenOrca	Mistral-7B	Random 5B tokens	0.621	0.615	0.798	6.600
			Uniform 5B tokens	0.630	0.588	0.782	6.750
			ClusterClip	0.643	0.587	0.814	6.900
				MATH	GSM8K	MMLU	BBH
	Proof-Pile-2	LLaMA2-7B	Random 20B tkn	0.065	0.256	0.488	0.418
			Uniform 20B tkn	0.076	0.260	0.500	0.429
			ClusterClip	0.079	0.248	0.511	0.428
				MMLU	BBH	ARC	
QDIT [Bukharin & Zhao (2023)]	UltraChat	LLaMA-7B	Random 10K	0.321	0.332	0.583	
			QDIT 10K	0.361	0.321	0.607	
	LMSYS		Random 10K	0.331	0.326	0.602	
			QDIT 10K	0.373	0.325	0.614	
	Alpaca		Random 3K	0.362	0.303	0.617	
			QDIT 3K	0.355	0.304	0.620	
	Mixed		Random 10K	0.329	0.309	0.583	
			QDIT 10K	0.343	0.312	0.607	
	Dolly		Random 1K	0.281	0.273	0.594	
			QDIT 1K	0.338	0.303	0.598	
				DROP	LAMBADA	SciQ	
	UltraChat		Random 10K	0.262	0.698	0.854	
			QDIT 10K	0.267	0.698	0.868	
	LMSYS		Random 10K	0.251	0.685	0.867	
			QDIT 10K	0.264	0.693	0.850	
	Alpaca		Random 3K	0.263	0.716	0.870	
			QDIT 3K	0.270	0.697	0.841	
	Mixed		Random 10K	0.203	0.681	0.841	
			QDIT 10K	0.260	0.697	0.898	
	Dolly		Random 1K	0.173	0.717	0.807	
			QDIT 1K	0.226	0.723	0.806	
				BBH	DROP	MMLU	HumanEval
DQ [Zhou et al. (2023)]	Alpaca	LLaMA-7B	Full	0.329	0.263	0.416	0.100
			20%	0.327	0.267	0.398	0.092
			2%	0.329	0.276	0.366	0.085

highlight their corresponding facets without misinterpreting their mechanisms. Furthermore, we select the most recent methods that exploit open-sourced LLMs for verifying the effectiveness of their proposed data selection techniques. Such accessibility to public LLMs avoids the privacy issues and inference costs brought by requesting API services from close-sourced proprietary models.

Quality The quality of data directly impacts the effectiveness of model training. Quality control measures include data scoring, quality assessment, and more. In Tab. 2, we have summarized the results of different methods focusing on data quality. In the table, we list the data used by different methods and the proportion/size of the data selected. It can be seen that the method of selecting data based on quality can match the results of training with full data even under the data-poor regime. They are also superior to the results of randomly selecting a subset from the original dataset. In the table, WK stands for World Knowledge, CR stands for Commonsense Reasoning, LU stands for Language Understanding, SPS stands for Symbolic Problem Solving, and RC stands for Reading Comprehension.

Diversity Data engineering enhances the generalization ability of models by improving the diversity of datasets. Such improved diversity may be implemented via encompassing data from different sources, with varying features, and of distinct distributions. Researches indicate that it is insufficient to merely select datasets that are similar to the target data from the downstream tasks. Tab. 3 demonstrates the importance of diversity in data selection. Compared with random selection and uniform selection, the scheme of selecting data with diversity criteria is superior. In addition, compared to only selecting high-quality data, the criteria that combine quality and diversity can achieve better performance than simply selecting high-quality data.

Importance It is non-trivial to accurately identify and utilize datapoints that significantly affect model performance As shown in Tab. 4, the importance-based data selection approaches often maximize the final performance by integrating the data selection and model optimization processes. To address the challenges of potential huge computation overhead brought by the implementation framework, they propose to use instance-wise data impact and propose efficient parameterization. Moreover, by performing importance resampling in the feature space that depicts geometric structures, they select examples of both high importance and similarity to the target distribution, thereby enhancing the performance on the target tasks. Existing

work has confirmed that the importance sampling-based approach can effectively improve the performance on the target tasks with enhanced capabilities of LLMs.

Hybrid Selection It is noteworthy that many data selection methods such as LIFT [Xu et al. (2023b)], FL [Bhatt et al. (2024)], DEITA [Liu et al. (2023b)], and QDIT [Bukharin & Zhao (2023)], attempted to combine multiple aspects of data assessment into their selection pipelines. Most of them emphasize the overall assessment of instruction data in that the definitions of "good data" are varied under different scenarios in the era of LLMs. A unitary measurement would inevitably cause a biased selection of datapoints and thereafter leads to the degraded performance of LLMs. Technically, existing hybrid selection methods can be categorized into: 1) parallel setups, and 2) sequential setups. For the former, an operation of maximization or weighted sum is performed on scores or proxy indicators (i.e., losses and gradients) from multiple aspects (e.g., quality and diversity) [Xu et al. (2023b); Bhatt et al. (2024); Bukharin & Zhao (2023)]. For the latter, a sequential setup of quality-, diversity-, or importance-oriented selection techniques is performed step-by-step for hierarchical, stratified filtering [Liu et al. (2023b)]. Comparatively, the parallel setups allow dynamic trade-offs between multiple data aspects by adjusting the aggregation operations. The sequential setups, on the other hand, fail to retrieve the candidates that are filtered out in the preceding quality control steps even if those candidates are of high importance or variety. Therefore, it would be preferred to develop hybrid assessment techniques that simultaneously weigh quality, diversity, and importance. Furthermore, we notice that the importance-based assessment is often overlooked in existing hybrid approaches, implying that the investigation of integrating importance with quality and diversity is of high potentials for future studies.

Distinctions and Connections The methods that are respectively reported in Tabs. 2, 3, and 4 share the similar philosophy of data assessment and selection but differ in the detailed implementations.

For the quality-based selection, most of existing methods like IFD [Li et al. (2023a)], PPL [Ankner et al. (2024)], AutoDS [Zhang et al. (2024c)], and QuRator [Wettig et al. (2024)] mainly use the model-based indicators to measure the quality of each datapoint. However, their indicators are derived from different perspectives such as the perplexity of LLMs and the predicted quality score of a regression model. Both the Alpagasus [Chen et al. (2023b)] and BSDetector [Chen & Mueller (2024)] mainly employ ChatGPT scoring for quality evaluation. Alpagasus directly prompts GPT to score and filter out datapoints, but BSDetector considers both the consistency of the generated responses from GPT and the quality of the ground-truth response. On the other hand, LIFT [Xu et al. (2023b)], InstructionMining [Cao et al. (2023)], and FL [Bhatt et al. (2024)] develop composite methods that involve hand-crafted or model-based indicators, together with GPT scoring, for quality verification of the selected datapoints.

For the diversity-based selection, both DEITA [Liu et al. (2023b)] and ClusterClip [Shao et al. (2024)] adopt the geometry-based measurement where the geometric structure of datapoints is maintained via clustering. DEITA measures the individual-level relationship between one datapoint and its closest neighbor for sampling while ClusterClip simply performs the cluster-level uniform sampling with the quantity limit. QDIT [Bukharin & Zhao (2023)] performs the iterative selection where each step only one most promising datapoint from the remaining dataset is selected to maximize the overall diversity of the selected subset. Despite the fact that DQ [Zhou et al. (2023)] also exploits iterative coreset sampling, it first divides the entire dataset into multiple subset bins and performs sampling across bins for aggregation and deduplication.

For the importance-based selection, DsDm [Engstrom et al. (2024)] and MATES [Yu et al. (2024)] employ model-based indicators to estimate the influence with datamodels. However, these two methods choose different implementations of datamodels. DsDm develops a linear model and approximates the loss of each datapoint via the TARK estimator while MATES directly leverages a small language model to learn to predict the individual influence of each datapoint. DSIR [Xie et al. (2023)], on the contrary, bypasses the need to maintain a specific datamodel. Instead, it estimates the importance of each datapoint by comparing the distributions of the entire selected subset with those of the validation set. Both Skill-it [Chen et al. (2024b)] and LESS [Xia et al. (2024a)] pinpoint the samples that resemble the most to the target set. Skill-it performs sampling via minimizing the evaluation loss while LESS utilizes the gradient similarity as a proxy.

Table 4: Experimental results of importance-based selection methods are directly cited from their papers.

Methods	Training Set	Model	Ratio/Size	Reported Results on Testing Set				
				COPA	OBQA	PIQA	CBT	Hellaswag
DsDm [Engstrom et al. (2024)]	C4	Chinchilla-optimal-1.3B	Random	0.620	0.334	0.689	0.864	0.449
			DsDm	0.630	0.312	0.690	0.882	0.423
				Winogrande	BoolQ	COQA	ARC-E	TriviaQA
			Random	0.522	0.549	0.188	0.448	0.037
			DsDm	0.511	0.580	0.255	0.476	0.071
MATES [Yu et al. (2024)]	C4	Pythia-410M		SciQ	ARC-E	ARC-C	LogiQA	
			Random 20%	0.641	0.402	0.256	0.247	
			MATES 20%	0.660	0.418	0.250	0.257	
		Pythia-1B	Random 20%	0.658	0.437	0.256	0.275	
			MATES 20%	0.673	0.449	0.259	0.287	
		Pythia-410M		OBQA	BoolQ	HellaSwag	PIQA	Winogrande
			Random 20%	0.294	0.589	0.397	0.671	0.506
			MATES 20%	0.308	0.606	0.410	0.687	0.527
		Pythia-1B	Random 20%	0.318	0.602	0.438	0.689	0.507
			MATES 20%	0.322	0.609	0.453	0.695	0.524
DSIR [Xie et al. (2023)]	The Pile	RoBERTa-Base (125M)		MNLI	QNLI	QQP	RTE	
			Random 51.2M	0.826	0.869	0.896	0.674	
			DSIR 51.2M	0.831	0.891	0.898	0.751	
				SST-2	MRPC	CoLA	STS-B	
			Random 51.2M	0.901	0.874	0.494	0.886	
			DSIR 51.2M	0.905	0.877	0.540	0.892	
Skill-it [Chen et al. (2024b)]	RedPajama	GPT-Neo-3B		ARC-C	ARC-E	BoolQ	COPA	
			Skill-it 1B	0.346	0.612	0.682	0.820	
			Uniform 1B	0.354	0.652	0.689	0.810	
			Skill-it 1B	0.349	0.617	0.686	0.810	
			Uniform 1B	0.353	0.624	0.677	0.800	
			Skill-it 1B	0.348	0.620	0.687	0.810	
			Uniform 1B	0.346	0.625	0.672	0.810	
				HellaSwag	LAMBADA	PIQA	Winogrande	
			Skill-it 1B	0.637	0.670	0.750	0.639	
			Uniform 1B	0.639	0.644	0.748	0.628	
			Skill-it 1B	0.639	0.667	0.752	0.632	
			Uniform 1B	0.638	0.659	0.755	0.639	
			Skill-it 1B	0.639	0.660	0.757	0.631	
			Uniform 1B	0.640	0.668	0.750	0.634	
LESS [Xia et al. (2024a)]	Mixed	LLaMA2-7B		MMLU	TYDIQA	BBH		
			Full	0.516	0.540	0.432		
			Random 5%	0.465	0.527	0.389		
			LESS 5%	0.502	0.562	0.415		
		LLaMA2-13B	Full	0.545	0.543	0.508		
			Random 5%	0.534	0.530	0.470		
			LESS 5%	0.540	0.546	0.506		
		Mistral-7B	Full	0.604	0.577	0.530		
			Random 5%	0.600	0.569	0.545		
			LESS 5%	0.618	0.603	0.560		

7 Future Directions: Challenges and Opportunities

In this section, we present the existing challenges and potential solutions to developing advanced data assessment and selection methods.

7.1 Benchmarking Instruction-Tuned LLMs

There exists a gap between the effectiveness of data selection and the reported performance on benchmarks. In existing researches, the ablation studies on the effectiveness of assessment and selection methods are often carried out by comparing the performance of LLMs fine-tuned with the selected and the full dataset. However, for coreset sampling methods that use losses and gradients as proxies for data quality, the downstream performance may not be positively correlated with the selection effectiveness. The reason behind is that the evaluation loss itself [Yang et al. (2022); Hoffmann et al. (2022); Kaplan et al. (2020)] is not informative enough for universal estimation of benchmark performance. [AI@Meta (2024)] demonstrates that the correlation between the negative log-likelihood loss on downstream tasks and the accuracy metrics should be modeled task-by-task and model-by-model. In the light of this statement, it is impractical to simply count on losses or gradients to pinpoint the most beneficial data for improving the downstream performance, let alone methods that try to predict the loss based on various indicators [Cao et al. (2023)]. Furthermore, even if the metrics are exhaustively computed for the selection of each sample, the gains brought by one sample might be limited in few tasks. Therefore, to comprehensively reflect the effectiveness of sample selection, the evaluation of instruction-tuned models should be accompanied by the specialised evaluation of the selected datapoints. For the former, all sorts of evaluation strategies have been proposed to precisely evaluate the LLMs [Melis et al. (2017); Chang et al. (2024); Xu et al. (2022); Liang et al. (2022)]. The multiple-choice QA tasks are not enlightening in judging if the instruction-tuned model truly understands the problem rather than simply memorizing the answer choices given the instruction context. For the later, a benchmark for documenting and comparing the statistics of the selected instruction-response

pairs in terms of quality, diversity, and importance needs to be constructed in the future. It would benefit the task-wise customized data selection according to the statistical indicators on such a benchmark.

Test set contamination should be considered during instruction data selection. For instruction tuning on publicly released pre-trained LLMs, it cannot be too careful to check the potential data leakage where the testing instructions are already modeled during pre-training [Rae et al. (2021); Li et al. (2023b); Magar & Schwartz (2022); Carlini et al. (2019); Marone & Van Durme (2024); Deng et al. (2023); Cao et al. (2024); Jiang et al. (2024b); Magar & Schwartz (2022)]. To improve the performance of pre-trained models on downstream tasks, datapoints (i.e., instruction-like conversations) are already added into the annealing phase of pre-training [AI@Meta (2024); Bilibili (2024); Yang et al. (2024); Bai et al. (2023)]. Therefore, potential risks of data contamination are raised for benchmarking the fine-tuned LLMs. To avoid the negative effect of data leakage on evaluation of the data assessment and selection, it is encouraged to follow [Li et al. (2023a)] to adopt the pre-trained model for experiencing the datapoints before fine-tuning. If the model exhibits overfitting behaviors (i.e., accurately generating the instruction part or producing the same answer choice even with permutation on the choice letters), data contamination is likely to exist and thereafter the testing set should be replaced. For future studies, it would be more reliable to decouple the evaluation of data selection and that of fine-tuned LLMs, where the performance consistency between these two evaluation results can be analyzed to rule out the possibility of contamination.

7.2 Unveiling the Definitions of Good Data

What signifies the most a good datapoint remains an open question. Unfortunately, there exists no unified criteria on discriminating "good" instructions from "bad" ones. Essentially, the definitions on the general data "quality" differ from task to task and domain to domain [Evans & Murshudov (2013); Flach (2012); Albalak et al. (2024)]. Although existing quality measurement methods can be categorized in terms of quality, diversity, and importance under the present study, they all exhibit more or less ad-hoc properties in methodology. First, studies on instruction tuning are often targeted at improving the performance of LLMs on downstream task. No matter whether these tasks are of general-purpose (e.g., common NLP tasks on leaderboard [Myrzakhan et al. (2024); Wolf et al. (2019)]) or domain-specific applications, such task-orientated data selection itself is only a "proxy" for exploring the underlying "quality" measurement. Especially for coreset sampling methods that directly employ the evaluation set or testing set for distribution matching or importance estimation, instructions that resemble the most to the testing set or bring about performance gains are judged as "good" data. However, such "good" data cannot be easily transferred to another LLM of completely different architecture and parameters. Each time the entire pipeline has to be enforced for a novel task, making it difficult to accumulate universally-acknowledged high-quality data for archiving. Second, each method has an individual quality evaluation system and very few of them ever tried to justify their design and interpret the philosophy behind. It is difficult to validate whether certain component of the selection pipeline can be replaced or removed for better serving a new task-of-interest.

Accordingly, further academic explorations include: 1) to present a more unified, generally applicable definitions on "good" datapoints in terms of fine-grained aspects, and 2) to improve interpretability and explanability of the selection pipeline beyond empirical design.

The expected model behavior retrospectively determines the trade-off between quality, diversity, and importance for data selection. The three aspects we used to categorize data assessment methods are actually overlapping with each other, where the "boundary" between two measuring dimension is often hard to explicitly defined. Under such circumstance, the definition of good data can be perceived as the weighted, biased mixture of quality, diversity, and importance. Existing methods are not flexible in dynamically adjusting the mixing weights to adapt to different downstream tasks. Instead, their priority order of the three dimensions is implicitly encoded into the selection of instructions. For instance, [Liu et al. (2023b)] emphasizes quality and importance equally by first establishing the relative ranking of all samples in both quality and complexity. The subset is formed by consecutively selecting the top-ranked samples in sequence, with diversity intervened via ruling out heavily homogenized examples. Such hard-coded, greedy treatment to quality, diversity, and importance is not applicable to scenarios where the behavior of LLMs is expected to cater to varied preference.

In general, the data assessment and selection methods that can adapt to the model requirement under different application scenarios are yet to be systematically developed. For generation tasks like role playing and creative writing, the preferred instruction tuning datapoints should be distinct from those for discriminative tasks like named entity recognition and sentiment analysis.

7.3 Scaling Up Datasets

The optimal scale of the selected subset becomes less explicit with the expansion of datasets. In the analysis of the disadvantages in exploiting the entire instruction dataset for alignment, putting aside the issue of long training time, we notice that the performance of fine-tuning the entire dataset might not be the optimal. There often exists a critical point of the best selection proportion, and such proportion varies from dataset to dataset. When more instruction datasets from diverse domains and tasks are incorporated, it becomes more difficult to nail down the best selection proportion for three main reasons. First, a large proportion of noise exists in the open datasets and few noisy samples can already cause tremendous negative impact on performance [Song et al. (2022)]. During the pre-processing of instruction dataset, noise can be unintentionally introduced in instruction preparation (e.g., missing context or system prompt), response generation (e.g., unverified or mismatched answers), format wrapping (e.g., invalid `JSON` and unresolved `code`), and text augmentation (e.g., synonym replacement and reorder of words). Second, for specialized tasks in "vertical" domains, the overfitting of specific prompts occurs [Ma et al. (2023)] when the diversity of input instructions is rather limited. Despite the accuracy and rationality of the instruction-response pairs, LLMs tend to overfit certain patterns of the input instruction rather than truly comprehend the task. Therefore, the increase of dataset size instead reduces the generalization of trained LLMs with lower instruction following capabilities. Third, the forgetting [Zhang & Wu (2024); Jin & Ren (2024b); Wang et al. (2024b)] becomes a severe problem when more instruction datasets are introduced without setting a proper re-playing schedule of pre-training or previously visited instruction tuning datapoints [Parmar et al. (2024); Jin et al. (2021); Ibrahim et al. (2024)]. The skill cultivation of a LLM on any new instruction task heavily relies on its preceding skills acquired during pre-training or previous fine-tuning. Consequently, the expansion of samples for high-level skills would "dilute" those for low-level skills and degrade performance.

To sum up, with the dataset scaling up, fine-tuning with the selected subset instead of the entire set becomes a must-have strategy. To help determine the optimal selection ratio, we suggest the following three guidelines: 1) One may first develop a complex quality measure scheme that uses both indicators and human verification to estimate the noise percentage of each constituting dataset. Without lose of generality, random sampling can be performed to accelerate quality measurement. To combat noise, a lower ratio (i.e., smaller S_b) should be considered for data selection from the noisy S. 2) To combat overfitting, both the diversity of datapoints within and across datasets should be emphasized. A higher keeping proportion should be established for datasets with diverse instruction styles, prompt formations, and response patterns, which helps improve the model's instruction following capabilities. 3) For continual fine-tuning, datasets that share similar distributions with pre-training and previous fine-tuning datapoints should be kept to fight against forgetting. The optimal selection ratio and proportion for each dataset is built upon the meticulous and thorough analysis on each instruction dataset, and therefore case-by-case adjustment is needed. For future studies, one may investigate the automation of assessment and selection recipe to minimize the human intervention.

The optimization of a scalable pipeline for data assessment and selection is of urgent need. In consideration of the cost of building human-labeled and human-verified instruction tuning datasets, methods that employ powerful LLMs like GPT4 for instruction synthesis [Bradley et al. (2023); Li et al. (2023f); Xu et al. (2023a); Li et al. (2024a); Zhao et al. (2024a); Dong et al. (2024)] have gained increasing attention. The synthetic instructions proliferate cost-effectively with fine-grained control of characteristics such as difficulty and style. Therefore, it is expected to witness a surge of datapoints (e.g., tens or even hundreds millions) in the short future. In that case, datasets of such quantity pose a significant challenge to the robustness, efficiency, and precision of the selection methods. Previous studies like DSIM [Xie et al. (2023)] demonstrated that cheap approximation of features by bag-of-n-grams achieves similar performance but requires much less computing resources. For future research, one may draw inspiration from the data deduplication and filtering techniques in handling billions of pre-training tokens. Especially for the measurement of diversity, the computing of embedding-based pairwise similarity and clustering can be greatly reduced with simplified representations. In

addition, the hierarchical philosophy [Hmida et al. (2016); Talavera (1999); Cabezas et al. (2023); Ran et al. (2023)] might be a promising approach to select data from coarse-grained to fine-grained structures. One may apply the devide-and-conquer strategy to recursively handle each subset of the instruction dataset, limiting the peak resource consumption under budget.

7.4 Scaling Up LLMs

The cost-efficiency of data assessment and selection diminishes with larger LLMs involved in the pipeline. The model-based indicators and coreset sampling methods often require the language model itself to be involved for computation of metrics [Li et al. (2023a)], losses [Chen et al. (2024b)], and gradients [Xia et al. (2024a)]. With the increase of model sizes, it becomes more and more cumbersome to implement the entire pipeline for quality measurement and selection. To expedite the process, one important direction for future study is to develop proper efficient proxy models. Small proxy models have been successfully applied in accelerated fine-tuning of language models [Hoffmann et al. (2022); Liu et al. (2024a)], filtering datasets by perplexity [Ankner et al. (2024)], intervention on retrieval-augmented generation [Tan et al. (2024b)], and performance prediction [Anugraha et al. (2024); Ngu et al. (2024)]. Such proxy models often share the same architecture design with the LLMs under development but own much less parameters. The scaling law [Kaplan et al. (2020)] confirms the expected consistent behavior between data quantity and model scale, providing practical guidelines on the development of such proxy LLMs.

On the other hand, under the context of data evaluation, it calls upon on rethinking of traditional machine learning techniques such as efficient optimization tricks and dimensionality reduction approaches. For example, in the assessment of loss-based datapoint influence [Feldman & Zhang (2020)], the exhaustive measurement on the marginal performance by moving-each-sample-out and model re-training can be simply approximated by iterative batch-wise sampling tricks with a greedy principle behind. For efficient assessment, PCA [Xu et al. (2023b)] and random projection [Xia et al. (2024a); Park et al. (2023)] are popular choices for obtaining low-rank representations of embeddings and gradients, which facilitates not only metric computation but also storing of datapoints.

The marginal benefits of instruction tuning diminishes with increasing size of LLMs for knowledge supplement. Recent studies on the effectiveness of instruction tuning in injecting task-specific or domain-specific knowledge into LLMs [Shi & Lipani (2023); Goyal et al. (2023); Zhang et al. (2024b); Yıldız et al. (2024)] show that the stand-alone instruction tuning might not be the most appropriate method. Compared with strategies like continual pre-training [Cossu et al. (2024); Ke (2024); Cossu et al. (2024)] and instruction modeling [Lou et al. (2024); Cheng et al. (2024); Wang et al. (2022); Xu et al. (2024); Shi et al. (2024)], instruction tuning counts the response sequences for loss computation without sufficient perception of instruction context. For specialized domains like medicine, finance, and laws, if the pre-trained LLMs are in lack of the prerequisite knowledge, the instruction tuning cannot properly activates the parameterized "memory" for alignment but only causes overfitting of the given prompt. In that case, the benefits of data selection are limited with poor generalizability.

Another noteworthy phenomenon in data assessment and selection studies is that due to limited budgets of computing resources, most of the experiments are performed on LLMs of small and moderate size (e.g., less than 7B) to validate the effectiveness of the quality measurement and the selection strategy. Small pre-trained LLMs, by their nature of small parameter size, are more sensitive to the instruction datasets during fine-tuning or continual learning [Schick & Schütze (2020); Yıldız et al. (2024)]. They exhibit the most significant rates of both forggeting (old knowledge) and learning (new knowledge). In the light of such statement, small LLMs tend to sacrifice the task-irrelevant knowledge in return for rapid adaptation towards novel domains and tasks. The selected datasets by various quality measures can impose immediate effect on the parameters of small LLMs, but may weaken on those of huge ones. It remains unknown whether the same quality measurement and data selection pipeline can achieve similar performance gains on both small and large LLMs. For future research of data assessment and selection, extensive experiments are required to validate their efficiency on huge LLMs (e.g., 70B and 405B) [AI@Meta (2024)] and LLMs of mixture-of-experts (MoE) architectures (e.g., Mixtral 8x22B) [Jiang et al. (2024a)].

In consideration of the pre-training corpus, extremely large LLMs already experienced a vast amount of multi-lingual, multi-domain web texts during pre-training, and therefore the priority of the dimensions in data assessment (i.e., quality, diversity, and importance) differs from small LLMs. The association between the model scale and the data selection criteria is yet to be studied.

7.5 Validating the Bias and Fairness of Instruction-tuned LLMs

As mentioned in the Sec. 1.3, most existing data selection methods fail to specifically study the dataset bias for consideration of fairness. It is noted that for domain-specific tasks such as solving programming and maths problems, the data bias might not be that severe. However, for general question-answering, recommendation, and creative writing tasks, even ChatGPT could produce biased answers that are intolerable in domains like education [Doan et al. (2024); Zhang et al. (2023b); Chisca et al. (2024); Gao et al. (2024)]. To improve the fairness of LLMs, it is necessary to take the measurement of data bias into serious consideration. Future work includes: 1) the integration of embedding-based (e.g., WEAT [Caliskan et al. (2017)], SEAT [May et al. (2019)]), probability-based (e.g., DisCo [Webster et al. (2020)], CrowS-Pairs Score [Nangia et al. (2020)]), and generated texts-based (e.g., Co-Occurrence Bias Score [Bordia & Bowman (2019)], Full Gen Bias [Smith et al. (2022)]) techniques for fairness evaluation of datapoints [Gallegos et al. (2024)], and 2) the construction of bias and fairness benchmarks [Jiao et al. (2024)] under various domains and tasks as supplementary kits to the current evaluation pipeline that only focuses on testing the capability of LLMs.

8 Conclusion

In this study, we have thoroughly examined the state-of-the-art data assessment and selection methods for instruction tuning of LLMs. The present review presents a unified organization and categorizes these methods in terms of measuring dimensionality: quality, diversity, and importance. In each dimensionality, we outline the representative strategies in details and describe the factors to consider when selecting data for instruction tuning. Furthermore, we report the performance of typical data selection methods and provide discussions on the comparison between these methods. Last but not least, the existing challenges and potential solutions for future studies are summarized in hope for benefitting the research community.

References

Amro Abbas, Kushal Tirumala, Dániel Simig, Surya Ganguli, and Ari S Morcos. Semdedup: Data-efficient learning at web-scale through semantic deduplication. *arXiv preprint arXiv:2303.09540*, 2023.

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Alessandro Achille, Michael Lam, Rahul Tewari, Avinash Ravichandran, Subhransu Maji, Charless C Fowlkes, Stefano Soatto, and Pietro Perona. Task2vec: Task embedding for meta-learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6430–6439, 2019.

Sandesh Adhikary and Byron Boots. Sampling over riemannian manifolds using kernel herding. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 3646–3653. IEEE, 2022.

Naman Agarwal, Brian Bullins, and Elad Hazan. Second-order stochastic optimization in linear time. *stat*, 1050:15, 2016.

Sharat Agarwal, Himanshu Arora, Saket Anand, and Chetan Arora. Contextual diversity for active learning. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XVI 16*, pp. 137–153. Springer, 2020.

AI@Meta. Llama 3 model card. 2024. URL https://github.com/meta-llama/llama3/blob/main/MODEL_CARD.md.

Ahmed Alaa and Mihaela Van Der Schaar. Discriminative jackknife: Quantifying uncertainty in deep learning via higher-order influence functions. In *International Conference on Machine Learning*, pp. 165–174. PMLR, 2020.

Alon Albalak, Yanai Elazar, Sang Michael Xie, Shayne Longpre, Nathan Lambert, Xinyi Wang, Niklas Muennighoff, Bairu Hou, Liangming Pan, Haewon Jeong, et al. A survey on data selection for language models. *arXiv preprint arXiv:2402.16827*, 2024.

Alexandre Alcoforado, Thomas Palmeira Ferraz, Lucas Hideki Okamura, Israel Campos Fama, Arnold Moya Lavado, Bárbara Dias Bueno, Bruno Veloso, and Anna Helena Reali Costa. From random to informed data selection: A diversity-based approach to optimize human annotation and few-shot learning. *arXiv preprint arXiv:2401.13229*, 2024.

Zachary Ankner, Cody Blakeney, Kartik Sreenivasan, Max Marion, Matthew L Leavitt, and Mansheej Paul. Perplexed by perplexity: Perplexity-based data pruning with small reference models. *arXiv preprint arXiv:2405.20541*, 2024.

Antonis Antoniades, Xinyi Wang, Yanai Elazar, Alfonso Amayuelas, Alon Albalak, Kexun Zhang, and William Yang Wang. Generalization vs memorization: Tracing language models' capabilities back to pretraining data. *arXiv preprint arXiv:2407.14985*, 2024.

David Anugraha, Genta Indra Winata, Chenyue Li, Patrick Amadeus Irawan, and En-Shiun Annie Lee. Proxylm: Predicting language model performance on multilingual tasks via proxy models. *arXiv preprint arXiv:2406.09334*, 2024.

Walter E. Arnoldi. The principle of minimized iterations in the solution of the matrix eigenvalue problem. *Quarterly of Applied Mathematics*, 9:17–29, 1951. URL `https://api.semanticscholar.org/CorpusID:115852469`.

Kyriakos Axiotis, Vincent Cohen-Addad, Monika Henzinger, Sammy Jerome, Vahab Mirrokni, David Saulpic, David Woodruff, and Michael Wunder. Data-efficient learning via clustering-based sensitivity sampling: Foundation models and beyond. *arXiv preprint arXiv:2402.17327*, 2024.

Francis Bach and Michael Jordan. Learning spectral clustering. *Advances in neural information processing systems*, 16, 2003.

Juhan Bae, Wu Lin, Jonathan Lorraine, and Roger Grosse. Training data attribution via approximate unrolled differentation. *arXiv preprint arXiv:2405.12186*, 2024.

Prafulla Bafna, Dhanya Pramod, and Anagha Vaidya. Document clustering: Tf-idf approach. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, pp. 61–66. IEEE, 2016.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

Maria-Florina Balcan, Alina Beygelzimer, and John Langford. Agnostic active learning. In *Proceedings of the 23rd international conference on Machine learning*, pp. 65–72, 2006.

Lukas Balles, Giovanni Zappella, and Cédric Archambeau. Gradient-matching coresets for rehearsal-based continual learning. *arXiv preprint arXiv:2203.14544*, 2022.

Sayan Bandyapadhyay and Kasturi Varadarajan. On variants of k-means clustering. *arXiv preprint arXiv:1512.02985*, 2015.

Samyadeep Basu, Philip Pope, and Soheil Feizi. Influence functions in deep learning are fragile. *arXiv preprint arXiv:2006.14651*, 2020.

Carlo Batini, Cinzia Cappiello, Chiara Francalanci, and Andrea Maurino. Methodologies for data quality assessment and improvement. *ACM computing surveys (CSUR)*, 41(3):1–52, 2009.

John C Begeny and Diana J Greene. Can readability formulas be used to successfully gauge difficulty of reading materials? *Psychology in the Schools*, 51(2):198–215, 2014.

Yonatan Belinkov and James Glass. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72, 2019.

Yves Bestgen. Measuring lexical diversity in texts: The twofold length problem. *Language Learning*, 2023.

Gantavya Bhatt, Yifang Chen, Arnav M Das, Jifan Zhang, Sang T Truong, Stephen Mussmann, Yinglun Zhu, Jeffrey Bilmes, Simon S Du, Kevin Jamieson, et al. An experimental design framework for label-efficient supervised finetuning of large language models. *arXiv preprint arXiv:2401.06692*, 2024.

Stella Biderman, Usvsn Prashanth, Lintang Sutawika, Hailey Schoelkopf, Quentin Anthony, Shivanshu Purohit, and Edward Raff. Emergent and predictable memorization in large language models. *Advances in Neural Information Processing Systems*, 36, 2024.

Bilibili. Index1.9b technical report. 2024. URL `https://github.com/bilibili/Index-1.9B`.

A Bjork. Least squares methods: Handbook of numerical analysis. *Finite Difference Methods Solutions of Equations in Rn*, 1, 1988.

David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022, 2003.

Shikha Bordia and Samuel R Bowman. Identifying and reducing gender bias in word-level language models. *arXiv preprint arXiv:1904.03035*, 2019.

Zalán Borsos, Mojmir Mutny, and Andreas Krause. Coresets via bilevel optimization for continual learning and streaming. *Advances in neural information processing systems*, 33:14879–14890, 2020.

Zalán Borsos, Mojmír Mutnỳ, Marco Tagliasacchi, and Andreas Krause. Data summarization via bilevel optimization. *Journal of Machine Learning Research*, 25(73):1–53, 2024.

Maximilian Böther, Viktor Gsteiger, Ties Robroek, and Ana Klimovic. Modyn: A platform for model training on dynamic datasets with sample-level data selection. *arXiv preprint arXiv:2312.06254*, 2023.

Herbie Bradley, Andrew Dai, Hannah Teufel, Jenny Zhang, Koen Oostermeijer, Marco Bellagente, Jeff Clune, Kenneth Stanley, Grégory Schott, and Joel Lehman. Quality-diversity through ai feedback. *arXiv preprint arXiv:2310.13032*, 2023.

Jonathan Brophy, Zayd Hammoudeh, and Daniel Lowd. Adapting and evaluating influence-estimation methods for gradient-boosted decision trees. *Journal of Machine Learning Research*, 24(154):1–48, 2023.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Quang Vu Bui, Karim Sayadi, Soufian Ben Amor, and Marc Bui. Combining latent dirichlet allocation and k-means for documents clustering: effect of probabilistic based distance measures. In *Intelligent Information and Database Systems: 9th Asian Conference, ACIIDS 2017, Kanazawa, Japan, April 3-5, 2017, Proceedings, Part I 9*, pp. 248–257. Springer, 2017.

Alexander Bukharin and Tuo Zhao. Data diversity matters for robust instruction tuning. *arXiv preprint arXiv:2311.14736*, 2023.

John Byabazaire, Gregory O'Hare, and Declan Delaney. Data quality and trust: A perception from shared data in iot. In *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6. IEEE, 2020.

Luben MC Cabezas, Rafael Izbicki, and Rafael B Stern. Hierarchical clustering: Visualization, feature importance and model selection. *Applied Soft Computing*, 141:110303, 2023.

Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186, 2017.

Ricardo JGB Campello, Davoud Moulavi, and Jörg Sander. Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, pp. 160–172. Springer, 2013.

Jialun Cao, Wuqi Zhang, and Shing-Chi Cheung. Concerned with data contamination? assessing countermeasures in code language model. *arXiv preprint arXiv:2403.16898*, 2024.

Kris Cao and Stephen Clark. Latent variable dialogue models and their diversity. *arXiv preprint arXiv:1702.05962*, 2017.

Yihan Cao, Yanbin Kang, and Lichao Sun. Instruction mining: High-quality instruction data selection for large language models. *arXiv preprint arXiv:2307.06290*, 2023.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX security symposium (USENIX security 19)*, pp. 267–284, 2019.

Patricia L Carrell. Readability in esl. 1987.

Jeanne Sternlicht Chall and Edgar Dale. Readability revisited: The new dale-chall readability formula. *Edgar Brookline Books*, 1995.

Chi-Min Chan, Weize Chen, Yusheng Su, Jianxuan Yu, Wei Xue, Shanghang Zhang, Jie Fu, and Zhiyuan Liu. Chateval: Towards better llm-based evaluators through multi-agent debate. *arXiv preprint arXiv:2308.07201*, 2023.

KHR Chan, Y Yu, C You, H Qi, J Wright, and YR Ma. a white-box deep network from the principle of maximizing rate reduction. arxiv. 2021. *arXiv preprint arXiv:2105.10446*, 2021.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM Transactions on Intelligent Systems and Technology*, 15(3):1–45, 2024.

Daoyuan Chen, Yilun Huang, Zhijian Ma, Hesen Chen, Xuchen Pan, Ce Ge, Dawei Gao, Yuexiang Xie, Zhaoyang Liu, Jinyang Gao, et al. Data-juicer: A one-stop data processing system for large language models. In *Companion of the 2024 International Conference on Management of Data*, pp. 120–134, 2024a.

Hao Chen, Yiming Zhang, Qi Zhang, Hantao Yang, Xiaomeng Hu, Xuetao Ma, Yifan Yanggong, and Junbo Zhao. Maybe only 0.5% data is needed: A preliminary exploration of low training data instruction tuning. *arXiv preprint arXiv:2305.09246*, 2023a.

Jiuhai Chen and Jonas Mueller. Quantifying uncertainty in answers from any language model and enhancing their trustworthiness. 2023.

Jiuhai Chen and Jonas Mueller. Automated data curation for robust language model fine-tuning. *arXiv preprint arXiv:2403.12776*, 2024.

Lichang Chen, Shiyang Li, Jun Yan, Hai Wang, Kalpa Gunaratna, Vikas Yadav, Zheng Tang, Vijay Srinivasan, Tianyi Zhou, Heng Huang, et al. Alpagasus: Training a better alpaca with fewer data. *arXiv preprint arXiv:2307.08701*, 2023b.

Mayee Chen, Nicholas Roberts, Kush Bhatia, Jue Wang, Ce Zhang, Frederic Sala, and Christopher Ré. Skill-it! a data-driven skills framework for understanding and training language models. *Advances in Neural Information Processing Systems*, 36, 2024b.

Yutian Chen, Max Welling, and Alex Smola. Super-samples from kernel herding. *arXiv preprint arXiv:1203.3472*, 2012.

Yutian Chen, Luke Bornn, Nando De Freitas, Mareija Eskelin, Jing Fang, and Max Welling. Herded gibbs sampling. *The Journal of Machine Learning Research*, 17(1):263–291, 2016.

Daixuan Cheng, Yuxian Gu, Shaohan Huang, Junyu Bi, Minlie Huang, and Furu Wei. Instruction pre-training: Language models are supervised multitask learners. *arXiv preprint arXiv:2406.14491*, 2024.

Anshuman Chhabra, Peizhao Li, Prasant Mohapatra, and Hongfu Liu. " what data benefits my classifier?" enhancing model performance and interpretability through influence-based data selection. In *The Twelfth International Conference on Learning Representations*, 2024.

Andrei-Victor Chisca, Andrei-Cristian Rad, and Camelia Lemnaru. Prompting fairness: Learning prompts for debiasing large language models. In *Proceedings of the Fourth Workshop on Language Technology for Equality, Diversity, Inclusion*, pp. 52–62, 2024.

Zhibo Chu, Zichong Wang, and Wenbin Zhang. Fairness in large language models: A taxonomic survey. *ACM SIGKDD explorations newsletter*, 26(1):34–48, 2024.

Gui Citovsky, Giulia DeSalvo, Claudio Gentile, Lazaros Karydas, Anand Rajagopalan, Afshin Rostamizadeh, and Sanjiv Kumar. Batch active learning at scale. *Advances in Neural Information Processing Systems*, 34: 11933–11944, 2021.

Kevyn Collins-Thompson and Jamie Callan. Predicting reading difficulty with statistical language models. *Journal of the american society for information science and technology*, 56(13):1448–1462, 2005.

Benoît Colson, Patrice Marcotte, and Gilles Savard. An overview of bilevel optimization. *Annals of operations research*, 153:235–256, 2007.

Bradford R Connatser. Last rites for readability formulas in technical communication. *Journal of technical writing and communication*, 29(3):271–287, 1999.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. Free dolly: Introducing the world's first truly open instruction-tuned llm. *Company Blog of Databricks*, 2023.

William J Cook, William H Cunningham, William R Pulleyblank, and Alexander Schrijver. Combinatorial optimization. *Unpublished manuscript*, 10:75–93, 1994.

Gérard Cornuéjols, George Nemhauser, and Laurence Wolsey. The uncapicitated facility location problem. Technical report, Cornell University Operations Research and Industrial Engineering, 1983.

Andrea Cossu, Antonio Carta, Lucia Passaro, Vincenzo Lomonaco, Tinne Tuytelaars, and Davide Bacciu. Continual pre-training mitigates forgetting in language and vision. *Neural Networks*, pp. 106492, 2024.

Ian Covert, Wenlong Ji, Tatsunori Hashimoto, and James Zou. Scaling laws for the value of individual data points in machine learning. *arXiv preprint arXiv:2405.20456*, 2024.

M Covington and Joe D McFall. The moving-average type-token ratio. *Linguistics Society of America, Chicago, IL*, 2008.

Michael A Covington and Joe D McFall. Cutting the gordian knot: The moving-average type–token ratio (mattr). *Journal of quantitative linguistics*, 17(2):94–100, 2010.

Radu G Creţulescu, Daniel I Morariu, Macarie Breazu, and Daniel Volovici. Dbscan algorithm for document clustering. *International Journal of Advanced Statistics and IT&C for Economics and Life Sciences*, 9(1): 58–66, 2019.

Edgar Dale and Jeanne S Chall. The concept of readability. *Elementary English*, 26(1):19–26, 1949.

Dan Dan Friedman and Adji Bousso Dieng. The vendi score: A diversity evaluation metric for machine learning. *Transactions on machine learning research*, 2023.

Vu Minh Hoang Dang and Rakesh M Verma. Data quality in nlp: Metrics and a comprehensive taxonomy. In *International Symposium on Intelligent Data Analysis*, pp. 217–229. Springer, 2024.

Fredrik deBoer. Evaluating the comparability of two measures of lexical diversity. *System*, 47:139–145, 2014.

Chunyuan Deng, Yilun Zhao, Xiangru Tang, Mark Gerstein, and Arman Cohan. Investigating data contamination in modern benchmarks for large language models. *arXiv preprint arXiv:2311.09783*, 2023.

Dingsheng Deng. Dbscan clustering algorithm based on density. In *2020 7th international forum on electrical engineering and automation (IFEEA)*, pp. 949–953. IEEE, 2020.

Mingkai Deng, Bowen Tan, Zhengzhong Liu, Eric P Xing, and Zhiting Hu. Compression, transduction, and creation: A unified framework for evaluating natural language generation. *arXiv preprint arXiv:2109.06379*, 2021.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*, 2023.

Marcel Dix, Gianluca Manca, Kenneth Chigozie Okafor, Reuben Borrison, Konstantin Kirchheim, Divyasheel Sharma, Kr Chandrika, Deepti Maduskar, and Frank Ortmeier. Measuring the robustness of ml models against data quality issues in industrial time series data. In *2023 IEEE 21st International Conference on Industrial Informatics (INDIN)*, pp. 1–8. IEEE, 2023.

Thang Viet Doan, Zichong Wang, Nhat Nguyen Minh Hoang, and Wenbin Zhang. Fairness in large language models in three hours. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pp. 5514–5517, 2024.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305*, 2020.

Guanting Dong, Keming Lu, Chengpeng Li, Tingyu Xia, Bowen Yu, Chang Zhou, and Jingren Zhou. Self-play with execution feedback: Improving instruction-following capabilities of large language models. *arXiv preprint arXiv:2406.13542*, 2024.

Hanze Dong, Wei Xiong, Deepanshu Goyal, Yihan Zhang, Winnie Chow, Rui Pan, Shizhe Diao, Jipeng Zhang, Kashun Shum, and Tong Zhang. Raft: Reward ranked finetuning for generative foundation model alignment. *arXiv preprint arXiv:2304.06767*, 2023.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. Unified language model pre-training for natural language understanding and generation. *Advances in neural information processing systems*, 32, 2019.

Wei Dong, Charikar Moses, and Kai Li. Efficient k-nearest neighbor graph construction for generic similarity measures. In *Proceedings of the 20th international conference on World wide web*, pp. 577–586, 2011.

Jiawei Du, Qin Shi, and Joey Tianyi Zhou. Sequential subset matching for dataset distillation. *Advances in Neural Information Processing Systems*, 36, 2024.

Nan Du, Yingyu Liang, Maria Balcan, and Le Song. Influence function learning in information diffusion networks. In *International Conference on Machine Learning*, pp. 2016–2024. PMLR, 2014.

Qianlong Du, Chengqing Zong, and Jiajun Zhang. Mods: Model-oriented data selection for instruction tuning. *arXiv preprint arXiv:2311.15653*, 2023.

Wenchao Du and Alan W Black. Boosting dialog response generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

William H Dubay. The principles of readability. impact information. *Costa Mesa, CA*, 2004.

Yann Dubois, Chen Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy S Liang, and Tatsunori B Hashimoto. Alpacafarm: A simulation framework for methods that learn from human feedback. *Advances in Neural Information Processing Systems*, 36, 2024.

Sergey Edunov, Alexei Baevski, and Michael Auli. Pre-trained language model representations for language generation. *arXiv preprint arXiv:1903.09722*, 2019.

Lisa Ehrlinger and Wolfram Wöß. A survey of data quality measurement and monitoring tools. *Frontiers in big data*, 5:850611, 2022.

Yuval Eldar, Michael Lindenbaum, Moshe Porat, and Yehoshua Y Zeevi. The farthest point strategy for progressive image sampling. *IEEE transactions on image processing*, 6(9):1305–1315, 1997.

Logan Engstrom, Axel Feldmann, and Aleksander Madry. Dsdm: Model-aware dataset selection with datamodels. *arXiv preprint arXiv:2401.12926*, 2024.

Ayse Erkan and Yasemin Altun. Semi-supervised learning via generalized maximum entropy. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 209–216. JMLR Workshop and Conference Proceedings, 2010.

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. Understanding dataset difficulty with $\mathcal{V}$-usable information. In *International Conference on Machine Learning*, pp. 5988–6008. PMLR, 2022.

Philip R Evans and Garib N Murshudov. How good are my data and what is the resolution? *Acta Crystallographica Section D: Biological Crystallography*, 69(7):1204–1214, 2013.

Dante Everaert and Christopher Potts. Gio: Gradient information optimization for training dataset selection. *arXiv preprint arXiv:2306.11670*, 2023.

Reza Zanjirani Farahani and Masoud Hekmatfar. *Facility location: concepts, models, algorithms and case studies.* Springer Science & Business Media, 2009.

Dan Feldman and Michael Langberg. A unified framework for approximating and clustering data. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pp. 569–578, 2011.

Vitaly Feldman. Does learning require memorization? a short tale about a long tail. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pp. 954–959, 2020.

Vitaly Feldman and Chiyuan Zhang. What neural networks memorize and why: Discovering the long tail via influence estimation. *Advances in Neural Information Processing Systems*, 33:2881–2891, 2020.

Mariano Felice and Lucia Specia. Linguistic features for quality estimation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pp. 96–103, 2012.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. Language-agnostic bert sentence embedding. *arXiv preprint arXiv:2007.01852*, 2020.

Lijun Feng, Martin Jansche, Matt Huenerfauth, and Noémie Elhadad. A comparison of features for automatic readability assessment. In *Coling 2010: Posters*, pp. 276–284, 2010.

Steven Y Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard Hovy. A survey of data augmentation approaches for nlp. *arXiv preprint arXiv:2105.03075*, 2021.

Peter Flach. *Machine learning: the art and science of algorithms that make sense of data.* Cambridge university press, 2012.

Rudolph Flesch. A new readability yardstick. *Journal of applied psychology*, 32(3):221, 1948.

Thomas François. La lisibilité computationnelle: un renouveau pour la lisibilité du français langue première et seconde? *ITL-International Journal of Applied Linguistics*, 160(1):75–99, 2010.

Thomas François. *Les apports du traitement automatique du langage à la lisibilité du français langue étrangère*. PhD thesis, Ph. D. thesis, Université Catholique de Louvain. Thesis Supervisors: Cédrick . . . , 2011.

Thomas François and Cédrick Fairon. An "ai readability" formula for french as a foreign language. In *Proceedings of the 2012 joint conference on empirical methods in Natural Language Processing and computational natural language learning*, pp. 466–477, 2012.

Thomas François and Eleni Miltsakaki. Do nlp and machine learning improve traditional readability formulas? In *Proceedings of the First Workshop on Predicting and Improving Text Readability for target reader populations*, pp. 49–57, 2012.

Isabel O Gallegos, Ryan A Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K Ahmed. Bias and fairness in large language models: A survey. *Computational Linguistics*, pp. 1–79, 2024.

Rong Gao, Qin Ni, and Bingying Hu. Fairness of large language models in education. In *Proceedings of the 2024 International Conference on Intelligent Education and Computer Technology*, pp. 1, 2024.

Thomas George, César Laurent, Xavier Bouthillier, Nicolas Ballas, and Pascal Vincent. Fast approximate natural gradient descent in a kronecker-factored eigenbasis, 2021. URL https://arxiv.org/abs/1806.03884.

Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. In *International conference on machine learning*, pp. 2242–2251. PMLR, 2019.

Sreyan Ghosh, Chandra Kiran Reddy Evuru, Sonal Kumar, Deepali Aneja, Zeyu Jin, Ramani Duraiswami, Dinesh Manocha, et al. A closer look at the limitations of instruction tuning. *arXiv preprint arXiv:2402.05119*, 2024.

Sachin Goyal, Ananya Kumar, Sankalp Garg, Zico Kolter, and Aditi Raghunathan. Finetune like you pretrain: Improved finetuning of zero-shot vision models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 19338–19347, 2023.

Yves Grandvalet and Yoshua Bengio. Semi-supervised learning by entropy minimization. *Advances in neural information processing systems*, 17, 2004.

Hans-Rolf Gregorius and Elizabeth M Gillet. Generalized simpson-diversity. *Ecological Modelling*, 211(1-2): 90–96, 2008.

Roger Grosse, Juhan Bae, Cem Anil, Nelson Elhage, Alex Tamkin, Amirhossein Tajdini, Benoit Steiner, Dustin Li, Esin Durmus, Ethan Perez, Evan Hubinger, Kamilė Lukošiūtė, Karina Nguyen, Nicholas Joseph, Sam McCandlish, Jared Kaplan, and Samuel R. Bowman. Studying large language model generalization with influence functions, 2023. URL https://arxiv.org/abs/2308.03296.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, et al. Textbooks are all you need. *arXiv preprint arXiv:2306.11644*, 2023.

Robert Gunning. The technique of clear writing. *(No Title)*, 1952.

Chengcheng Guo, Bo Zhao, and Yanbing Bai. Deepcore: A comprehensive library for coreset selection in deep learning. In *International Conference on Database and Expert Systems Applications*, pp. 181–195. Springer, 2022.

Nitin Gupta, Shashank Mujumdar, Hima Patel, Satoshi Masuda, Naveen Panwar, Sambaran Bandyopadhyay, Sameep Mehta, Shanmukha Guttula, Shazia Afzal, Ruhi Sharma Mittal, et al. Data quality for machine learning tasks. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 4040–4041, 2021.

Kelvin Guu, Albert Webson, Ellie Pavlick, Lucas Dixon, Ian Tenney, and Tolga Bolukbasi. Simfluence: Modeling the influence of individual training examples by simulating training runs. *arXiv preprint arXiv:2303.08114*, 2023.

Nick Harvey and Samira Samadi. Near-optimal herding. In *Conference on Learning Theory*, pp. 1165–1182. PMLR, 2014.

Fangliang He and Xin-Sheng Hu. Hubbell's fundamental biodiversity parameter and the simpson diversity index. *Ecology Letters*, 8(4):386–390, 2005.

Jimming He, Sanjana Garg, and Jonas Mueller. How to detect bad data in your instruction tuning dataset (for better llm fine-tuning), 2024. URL `https://cleanlab.ai/blog/filter-llm-tuning-data/`.

Hmida Hmida, Sana Ben Hamida, Amel Borgi, and Marta Rukoz. Hierarchical data topology based selection for large scale learning. In *2016 Intl IEEE Conferences on Ubiquitous Intelligence & Computing, Advanced and Trusted Computing, Scalable Computing and Communications, Cloud and Big Data Computing, Internet of People, and Smart World Congress (UIC/ATC/ScalCom/CBDCom/IoP/SmartWorld)*, pp. 1221–1226. IEEE, 2016.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Anna Huang et al. Similarity measures for text document clustering. In *Proceedings of the sixth new zealand computer science research student conference (NZCSRSC2008), Christchurch, New Zealand*, volume 4, pp. 9–56, 2008.

Danqing Huang, Shuming Shi, Chin-Yew Lin, Jian Yin, and Wei-Ying Ma. How well do computers solve math word problems? large-scale dataset construction and evaluation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 887–896, 2016.

Hui Huang, Yingqi Qu, Jing Liu, Muyun Yang, and Tiejun Zhao. An empirical study of llm-as-a-judge for llm evaluation: Fine-tuned judge models are task-specific classifiers. *arXiv preprint arXiv:2403.02839*, 2024.

Ferenc Huszár and David Duvenaud. Optimally-weighted herding is bayesian quadrature. *arXiv preprint arXiv:1204.1664*, 2012.

Adam Ibrahim, Benjamin Thérien, Kshitij Gupta, Mats L Richter, Quentin Anthony, Timothée Lesort, Eugene Belilovsky, and Irina Rish. Simple and scalable strategies to continually pre-train large language models. *arXiv preprint arXiv:2403.08763*, 2024.

Abiodun M Ikotun, Absalom E Ezugwu, Laith Abualigah, Belal Abuhaija, and Jia Heming. K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. *Information Sciences*, 622:178–210, 2023.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Predicting predictions from training data. *arXiv preprint arXiv:2202.00622*, 2022.

Bogdan Ionescu, Mihai Lupu, Maia Rohm, Alexandru Lucian Gînsca, and Henning Müller. Datasets column: diversity and credibility for social images and image retrieval. *ACM SIGMultimedia Records*, 9(3):7–7, 2018.

Itay Itzhak, Gabriel Stanovsky, Nir Rosenfeld, and Yonatan Belinkov. Instructed to bias: Instruction-tuned language models exhibit emergent cognitive bias. *Transactions of the Association for Computational Linguistics*, 12:771–785, 2024.

Saachi Jain, Hadi Salman, Alaa Khaddaj, Eric Wong, Sung Min Park, and Aleksander Mądry. A data-based perspective on transfer learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3613–3622, 2023.

Sarthak Jain, Varun Manjunatha, Byron C Wallace, and Ani Nenkova. Influence functions for sequence tagging models. *arXiv preprint arXiv:2210.14177*, 2022.

Joel Jang, Seungone Kim, Seonghyeon Ye, Doyoung Kim, Lajanugen Logeswaran, Moontae Lee, Kyungjae Lee, and Minjoon Seo. Exploring the benefits of training expert language models over instruction tuning. In *International Conference on Machine Learning*, pp. 14702–14729. PMLR, 2023.

Scott Jarvis. Capturing the diversity in lexical diversity. *Language learning*, 63:87–106, 2013.

Scott Jarvis and M Daller. Defining and measuring lexical diversity. *Vocabulary knowledge: Human ratings and automated measures. Amsterdam, The Netherlands*, 2013.

Fred Jelinek, Robert L Mercer, Lalit R Bahl, and James K Baker. Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1):S63–S63, 1977.

Frederick Jelinek. Interpolated estimation of markov source parameters from sparse data. In *Proc. Workshop on Pattern Recognition in Practice, 1980*, 1980.

Hongjie Jia, Shifei Ding, Xinzheng Xu, and Ru Nie. The latest research progress on spectral clustering. *Neural Computing and Applications*, 24:1477–1486, 2014.

Ruoxi Jia, David Dao, Boxin Wang, Frances Ann Hubis, Nick Hynes, Nezihe Merve Gürel, Bo Li, Ce Zhang, Dawn Song, and Costas J Spanos. Towards efficient data valuation based on the shapley value. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1167–1176. PMLR, 2019.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023a.

Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024a.

Mingjian Jiang, Yangjun Ruan, Sicong Huang, Saifei Liao, Silviu Pitis, Roger Baker Grosse, and Jimmy Ba. Calibrating language models via augmented prompt ensembles. *Workshop on Challenges in Deployable Generative AI at International Conference on Machine Learning*, 2023b.

Minhao Jiang, Ken Ziyu Liu, Ming Zhong, Rylan Schaeffer, Siru Ouyang, Jiawei Han, and Sanmi Koyejo. Investigating data contamination for pre-training language models. *arXiv preprint arXiv:2401.06059*, 2024b.

Wenyu Jiang, Hao Cheng, Mingcai Chen, Chongjun Wang, and Hongxin Wei. Dos: Diverse outlier sampling for out-of-distribution detection. *arXiv preprint arXiv:2306.02031*, 2023c.

Wenyu Jiang, Zhenlong Liu, Zejian Xie, Songxin Zhang, Bingyi Jing, and Hongxin Wei. Exploring learning complexity for downstream data pruning. *arXiv preprint arXiv:2402.05356*, 2024c.

Zixuan Jiang, Jiaqi Gu, Mingjie Liu, and David Z Pan. Delving into effective gradient matching for dataset condensation. In *2023 IEEE International Conference on Omni-layer Intelligent Systems (COINS)*, pp. 1–6. IEEE, 2023d.

Tong Jiao, Jian Zhang, Kui Xu, Rui Li, Xi Du, Shangqi Wang, and Zhenbo Song. Enhancing fairness in llm evaluations: Unveiling and mitigating biases in standard-answer-based evaluations. In *Proceedings of the AAAI Symposium Series*, volume 4, pp. 56–59, 2024.

Xisen Jin and Xiang Ren. Demystifying forgetting in language model fine-tuning with statistical analysis of example associations. *arXiv preprint arXiv:2406.14026*, 2024a.

Xisen Jin and Xiang Ren. What will my model forget? forecasting forgotten examples in language model refinement. *arXiv preprint arXiv:2402.01865*, 2024b.

Xisen Jin, Dejiao Zhang, Henghui Zhu, Wei Xiao, Shang-Wen Li, Xiaokai Wei, Andrew Arnold, and Xiang Ren. Lifelong pretraining: Continually adapting language models to emerging corpora. *arXiv preprint arXiv:2110.08534*, 2021.

RC St John and Norman R Draper. D-optimality for regression designs: a review. *Technometrics*, 17(1): 15–23, 1975.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Firuz Kamalov. Kernel density estimation based sampling for imbalanced class distribution. *Information Sciences*, 512:1192–1201, 2020.

Nikhil Kandpal, Eric Wallace, and Colin Raffel. Deduplicating training data mitigates privacy risks in language models. In *International Conference on Machine Learning*, pp. 10697–10707. PMLR, 2022.

Feiyang Kang, Hoang Anh Just, Anit Kumar Sahu, and Ruoxi Jia. Performance scaling via optimal transport: Enabling data selection from partially revealed sources. *Advances in Neural Information Processing Systems*, 36, 2024.

Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine Piatko, Ruth Silverman, and Angela Y Wu. The analysis of a simple k-means clustering algorithm. In *Proceedings of the sixteenth annual symposium on Computational geometry*, pp. 100–109, 2000.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020.

Zixuan Ke. *Continual Learning with Language Models*. PhD thesis, University of Illinois at Chicago, 2024.

Susan Kemper. Measuring the inference load of a text. *Journal of educational psychology*, 75(3):391, 1983.

Kimmo Kettunen. Can type-token ratio be used to show morphological complexity of languages? *Journal of Quantitative Linguistics*, 21(3):223–245, 2014.

Kamran Khan, Saif Ur Rehman, Kamran Aziz, Simon Fong, and Sababady Sarasvady. Dbscan: Past, present and future. In *The fifth international conference on the applications of digital information and web technologies (ICADIWT 2014)*, pp. 232–238. IEEE, 2014.

Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, et al. Dynabench: Rethinking benchmarking in nlp. *arXiv preprint arXiv:2104.14337*, 2021.

Krishnateja Killamsetty, Sivasubramanian Durga, Ganesh Ramakrishnan, Abir De, and Rishabh Iyer. Grad-match: Gradient matching based data subset selection for efficient deep model training. In *International Conference on Machine Learning*, pp. 5464–5474. PMLR, 2021a.

Krishnateja Killamsetty, Durga Sivasubramanian, Ganesh Ramakrishnan, and Rishabh Iyer. Glister: Generalization based data subset selection for efficient and robust learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8110–8118, 2021b.

Krishnateja Killamsetty, Xujiang Zhao, Feng Chen, and Rishabh Iyer. Retrieve: Coreset selection for efficient and robust semi-supervised learning. *Advances in neural information processing systems*, 34:14488–14501, 2021c.

Walter Kintsch and Douglas Vipond. Reading comprehension and readability in educational practice and psychological theory. In *Perspectives on memory research*, pp. 329–365. Psychology Press, 2014.

John Kirchenbauer, Garrett Honke, Gowthami Somepalli, Jonas Geiping, Daphne Ippolito, Katherine Lee, Tom Goldstein, and David Andre. Lmd3: Language model data density dependence. *arXiv preprint arXiv:2405.06331*, 2024.

Andreas Kirsch. Does" deep learning on a data diet" reproduce? overall yes, but grand at initialization does not. *arXiv preprint arXiv:2303.14753*, 2023.

George R Klare. Assessing readability. *Reading research quarterly*, pp. 62–102, 1974.

George R Klare et al. *The measurement of readability*. Iowa State University Press Ames, 1963.

George R Klare et al. Readability. *Handbook of reading research*, 1:681–744, 1984.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pp. 1885–1894. PMLR, 2017.

Rik Koncel-Kedziorski, Subhro Roy, Aida Amini, Nate Kushman, and Hannaneh Hajishirzi. Mawps: A math word problem repository. In *Proceedings of the 2016 conference of the north american chapter of the association for computational linguistics: human language technologies*, pp. 1152–1157, 2016.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, et al. Openassistant conversations-democratizing large language model alignment. *Advances in Neural Information Processing Systems*, 36, 2024.

Jan Kremer, Kim Steenstrup Pedersen, and Christian Igel. Active learning with support vector machines. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 4(4):313–326, 2014.

Sandra Kublik and Shubham Saboo. *GPT-3: The Ultimate Guide to Building NLP Products with OpenAI API*. Packt Publishing Ltd, 2023.

Po-Nien Kung and Nanyun Peng. Do models really learn to follow instructions? an empirical study of instruction tuning. *arXiv preprint arXiv:2305.11383*, 2023.

Po-Nien Kung, Fan Yin, Di Wu, Kai-Wei Chang, and Nanyun Peng. Active instruction tuning: Improving cross-task generalization by training on prompt sensitive tasks. *arXiv preprint arXiv:2311.00288*, 2023.

Jeongyeol Kwon, Dohyun Kwon, Stephen Wright, and Robert D Nowak. A fully first-order method for stochastic bilevel optimization. In *International Conference on Machine Learning*, pp. 18083–18113. PMLR, 2023.

Yongchan Kwon and James Zou. Beta shapley: a unified and noise-reduced data valuation framework for machine learning. *arXiv preprint arXiv:2110.14049*, 2021.

Kristopher Kyle, Scott A Crossley, and Scott Jarvis. Assessing the validity of lexical diversity indices using direct judgements. *Language Assessment Quarterly*, 18(2):154–170, 2021.

Yi-An Lai, Xuan Zhu, Yi Zhang, and Mona Diab. Diversity, density, and homogeneity: Quantitative characteristic metrics for text collections. *arXiv preprint arXiv:2003.08529*, 2020.

Yrjo Lappalainen and Nikesh Narayanan. Aisha: A custom ai library chatbot using the chatgpt api. *Journal of Web Librarianship*, 17(3):37–58, 2023.

Stefan Larson, Anish Mahendran, Andrew Lee, Jonathan K Kummerfeld, Parker Hill, Michael A Laurenzano, Johann Hauswald, Lingjia Tang, and Jason Mars. Outlier detection for improved data quality and diversity in dialog systems. *arXiv preprint arXiv:1904.03122*, 2019.

Cosmin Lazar and Andrei Doncescu. Non negative matrix factorization clustering capabilities; application on multivariate image segmentation. In *2009 International Conference on Complex, Intelligent and Software Intensive Systems*, pp. 924–929. IEEE, 2009.

Ronan Le Bras, Swabha Swayamdipta, Chandra Bhagavatula, Rowan Zellers, Matthew Peters, Ashish Sabharwal, and Yejin Choi. Adversarial filters of dataset biases. In *International conference on machine learning*, pp. 1078–1088. Pmlr, 2020.

Guillaume Lecué and Shahar Mendelson. Regularization and the small-ball method i: sparse recovery. 2018.

Alycia Lee, Brando Miranda, Sudharsan Sundar, and Sanmi Koyejo. Beyond scale: the diversity coefficient as a data quality metric demonstrates llms are pre-trained on formally diverse data. *arXiv preprint arXiv:2306.13840*, 2023.

Daniel Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. *Advances in neural information processing systems*, 13, 2000.

Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. Deduplicating training data makes language models better. *arXiv preprint arXiv:2107.06499*, 2021.

Bohan Li, Yutai Hou, and Wanxiang Che. Data augmentation approaches in natural language processing: A survey. *Ai Open*, 3:71–90, 2022.

Haoran Li, Qingxiu Dong, Zhengyang Tang, Chaojun Wang, Xingxing Zhang, Haoyang Huang, Shaohan Huang, Xiaolong Huang, Zeqiang Huang, Dongdong Zhang, et al. Synthetic data (almost) from scratch: Generalized instruction tuning for language models. *arXiv preprint arXiv:2402.13064*, 2024a.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. *arXiv preprint arXiv:1510.03055*, 2015.

Ming Li, Yong Zhang, Zhitao Li, Jiuhai Chen, Lichang Chen, Ning Cheng, Jianzong Wang, Tianyi Zhou, and Jing Xiao. From quantity to quality: Boosting llm performance with self-guided data selection for instruction tuning. *arXiv preprint arXiv:2308.12032*, 2023a.

Ming Li, Yong Zhang, Shwai He, Zhitao Li, Hongyu Zhao, Jianzong Wang, Ning Cheng, and Tianyi Zhou. Superfiltering: Weak-to-strong data filtering for fast instruction-tuning. *arXiv preprint arXiv:2402.00530*, 2024b.

Na Li, Yiyang Qi, Chaoran Li, and Zhiming Zhao. Active learning for data quality control: A survey. *ACM Journal of Data and Information Quality*, 2024c.

Raymond Li, Loubna Ben Allal, Yangtian Zi, Niklas Muennighoff, Denis Kocetkov, Chenghao Mou, Marc Marone, Christopher Akiki, Jia Li, Jenny Chim, et al. Starcoder: may the source be with you! *arXiv preprint arXiv:2305.06161*, 2023b.

Xian Li, Ping Yu, Chunting Zhou, Timo Schick, Luke Zettlemoyer, Omer Levy, Jason Weston, and Mike Lewis. Self-alignment with instruction backtranslation. *arXiv preprint arXiv:2308.06259*, 2023c.

Yingji Li, Mengnan Du, Rui Song, Xin Wang, and Ying Wang. A survey on fairness in large language models. *arXiv preprint arXiv:2308.10149*, 2023d.

Yize Li, Pu Zhao, Xue Lin, Bhavya Kailkhura, and Ryan Goldhahn. Less is more: Data pruning for faster adversarial training. *arXiv preprint arXiv:2302.12366*, 2023e.

Yuchen Li, Ju Fan, Yanhao Wang, and Kian-Lee Tan. Influence maximization on social graphs: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 30(10):1852–1872, 2018.

Zhuoyan Li, Hangxiao Zhu, Zhuoran Lu, and Ming Yin. Synthetic data generation with large language models for text classification: Potential and limitations. *arXiv preprint arXiv:2310.07849*, 2023f.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *arXiv preprint arXiv:2211.09110*, 2022.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Jinkun Lin, Anqi Zhang, Mathias Lécuyer, Jinyang Li, Aurojit Panda, and Siddhartha Sen. Measuring the effect of training data on deep learning predictions via randomized experiments. In *International Conference on Machine Learning*, pp. 13468–13504. PMLR, 2022.

Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human falsehoods. *arXiv preprint arXiv:2109.07958*, 2021.

Xinyu Lin, Wenjie Wang, Yongqi Li, Shuo Yang, Fuli Feng, Yinwei Wei, and Tat-Seng Chua. Data-efficient fine-tuning for llm-based recommendation, 2024. URL https://arxiv.org/abs/2401.17197.

Alisa Liu, Xiaochuang Han, Yizhong Wang, Yulia Tsvetkov, Yejin Choi, and Noah A Smith. Tuning language models by proxy. *arXiv preprint arXiv:2401.08565*, 2024a.

Fuxiao Liu, Xiaoyang Wang, Wenlin Yao, Jianshu Chen, Kaiqiang Song, Sangwoo Cho, Yaser Yacoob, and Dong Yu. Mmc: Advancing multimodal chart understanding with large-scale instruction tuning. *arXiv preprint arXiv:2311.10774*, 2023a.

Qingyi Liu, Yekun Chai, Shuohuan Wang, Yu Sun, Keze Wang, and Hua Wu. On training data influence of gpt models. *arXiv preprint arXiv:2404.07840*, 2024b.

Wei Liu, Weihao Zeng, Keqing He, Yong Jiang, and Junxian He. What makes good data for alignment? a comprehensive study of automatic data selection in instruction tuning. *arXiv preprint arXiv:2312.15685*, 2023b.

Xiaoyu Liu, Paiheng Xu, Junda Wu, Jiaxin Yuan, Yifan Yang, Yuhang Zhou, Fuxiao Liu, Tianrui Guan, Haoliang Wang, Tong Yu, et al. Large language models and causal inference in collaboration: A comprehensive survey. *arXiv preprint arXiv:2403.09606*, 2024c.

Yang Liu, Jiahuan Cao, Chongyu Liu, Kai Ding, and Lianwen Jin. Datasets for large language models: A comprehensive survey. *arXiv preprint arXiv:2402.18041*, 2024d.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Ziche Liu, Rui Ke, Feng Jiang, and Haizhou Li. Take the essence and discard the dross: A rethinking on data selection for fine-tuning large language models. *arXiv preprint arXiv:2406.14115*, 2024e.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. The flan collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning*, pp. 22631–22648. PMLR, 2023.

Renze Lou, Kai Zhang, and Wenpeng Yin. Large language model instruction following: A survey of progresses and challenges. *Computational Linguistics*, pp. 1–10, 2024.

Keming Lu, Hongyi Yuan, Zheng Yuan, Runji Lin, Junyang Lin, Chuanqi Tan, Chang Zhou, and Jingren Zhou. # instag: Instruction tagging for analyzing supervised fine-tuning of large language models. In *The Twelfth International Conference on Learning Representations*, 2023a.

Yingzhou Lu, Minjie Shen, Huazheng Wang, Xiao Wang, Capucine van Rechem, and Wenqi Wei. Machine learning for synthetic data generation: a review. *arXiv preprint arXiv:2302.04062*, 2023b.

Zhaosong Lu and Sanyou Mei. First-order penalty methods for bilevel optimization. *SIAM Journal on Optimization*, 34(2):1937–1969, 2024.

Chengcheng Ma, Yang Liu, Jiankang Deng, Lingxi Xie, Weiming Dong, and Changsheng Xu. Understanding and mitigating overfitting in prompt tuning for vision-language models. *IEEE Transactions on Circuits and Systems for Video Technology*, 33(9):4616–4629, 2023.

Inbal Magar and Roy Schwartz. Data contamination: From memorization to exploitation. *arXiv preprint arXiv:2203.08242*, 2022.

Pratyush Maini, Saurabh Garg, Zachary Lipton, and J Zico Kolter. Characterizing datapoints via second-split forgetting. *Advances in Neural Information Processing Systems*, 35:30044–30057, 2022.

Jyoti Malhotra and Jagdish Bakal. A survey and comparative study of data deduplication techniques. In *2015 International Conference on Pervasive Computing (ICPC)*, pp. 1–5. IEEE, 2015.

David Malvern, Brian Richards, Ngoni Chipere, and Pilar Durán. *Lexical diversity and language development*. Springer, 2004.

David D Malvern and Brian J Richards. A new measure of lexical diversity. *British Studies in Applied Linguistics*, 12:58–71, 1997.

Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. When less is more: Investigating data pruning for pretraining llms at scale. *arXiv preprint arXiv:2309.04564*, 2023.

Marc Marone and Benjamin Van Durme. Data portraits: Recording foundation model training data. *Advances in Neural Information Processing Systems*, 36, 2024.

Michael Mathieu and Yann LeCun. Fast approximation of rotations and hessians matrices. *arXiv preprint arXiv:1404.7195*, 2014.

Vladimir Matlach, Diego Krivochen, and Jiri Milička. A method for comparison of general sequences via type-token ratio. *Language and Text: Data, models, information and applications. Amsterdam: John Benjamins*, pp. 37–54, 2021.

Chandler May, Alex Wang, Shikha Bordia, Samuel R Bowman, and Rachel Rudinger. On measuring social biases in sentence encoders. *arXiv preprint arXiv:1903.10561*, 2019.

Philip M McCarthy. *An assessment of the range and usefulness of lexical diversity measures and the potential of the measure of textual, lexical diversity (MTLD)*. PhD thesis, The University of Memphis, 2005.

Philip M McCarthy and Scott Jarvis. Mtld, vocd-d, and hd-d: A validation study of sophisticated approaches to lexical diversity assessment. *Behavior research methods*, 42(2):381–392, 2010.

Gábor Melis, Chris Dyer, and Phil Blunsom. On the state of the art of evaluation in neural language models. *arXiv preprint arXiv:1707.05589*, 2017.

Brando Miranda, Patrick Yu, Yu-Xiong Wang, and Sanmi Koyejo. The curse of low task diversity: On the failure of transfer learning to outperform maml and their empirical equivalence. *arXiv preprint arXiv:2208.01545*, 2022.

Swaroop Mishra and Bhavdeep Singh Sachdeva. Do we need to create big datasets to learn a task? In *Proceedings of SustaiNLP: Workshop on Simple and Efficient Natural Language Processing*, pp. 169–173, 2020.

Swaroop Mishra, Anjana Arunkumar, Bhavdeep Sachdeva, Chris Bryan, and Chitta Baral. Dqi: Measuring data quality in nlp. *arXiv preprint arXiv:2005.00816*, 2020a.

Swaroop Mishra, Anjana Arunkumar, Bhavdeep Sachdeva, Chris Bryan, and Chitta Baral. Dqi: A guide to benchmark evaluation. *arXiv preprint arXiv:2008.03964*, 2020b.

Niluthpol Chowdhury Mithun, Rameswar Panda, and Amit K Roy-Chowdhury. Construction of diverse image datasets from web collections with limited labeling. *IEEE Transactions on Circuits and Systems for Video Technology*, 30(4):1147–1161, 2019.

Sedir Mohammed, Hazar Harmouch, Felix Naumann, and Divesh Srivastava. Data quality assessment: Challenges and opportunities. *arXiv preprint arXiv:2403.00526*, 2024.

Robert C Moore and William Lewis. Intelligent selection of language model training data. In *Proceedings of the ACL 2010 conference short papers*, pp. 220–224, 2010.

David Mouillot and Alain Lepretre. A comparison of species diversity estimators. *Researches on Population Ecology*, 41:203–215, 1999.

Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

Aidar Myrzakhan, Sondos Mahmoud Bsharat, and Zhiqiang Shen. Open-llm-leaderboard: From multi-choice to open-style questions for llms evaluation, benchmark, and arena. *arXiv preprint arXiv:2406.07545*, 2024.

Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel R Bowman. Crows-pairs: A challenge dataset for measuring social biases in masked language models. *arXiv preprint arXiv:2010.00133*, 2020.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. *Advances in neural information processing systems*, 26, 2013.

Noel Ngu, Nathaniel Lee, and Paulo Shakarian. Diversity measures: Domain-independent proxies for failure in language model queries. In *2024 IEEE 18th International Conference on Semantic Computing (ICSC)*, pp. 176–182. IEEE, 2024.

Cuong V Nguyen, Alessandro Achille, Michael Lam, Tal Hassner, Vijay Mahadevan, and Stefano Soatto. Toward understanding catastrophic forgetting in continual learning. *arXiv preprint arXiv:1908.01091*, 2019.

Quan Nguyen and Adji Bousso Dieng. Quality-weighted vendi scores and their application to diverse experimental design. *arXiv preprint arXiv:2405.02449*, 2024.

Vu-Linh Nguyen, Mohammad Hossein Shaker, and Eyke Hüllermeier. How to measure uncertainty in uncertainty sampling for active learning. *Machine Learning*, 111(1):89–122, 2022.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*, 2019.

Björn Nieth, Thomas Altstidl, Leo Schwinn, and Björn Eskofier. Large-scale dataset pruning in adversarial training through data importance extrapolation. *arXiv preprint arXiv:2406.13283*, 2024.

Geir K Nilsen, Antonella Z Munthe-Kaas, Hans J Skaug, and Morten Brun. Efficient computation of hessian matrices in tensorflow. *arXiv preprint arXiv:1905.05559*, 2019.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744, 2022.

Rui Pan, Jipeng Zhang, Xingyuan Pan, Renjie Pi, Xiaoyu Wang, and Tong Zhang. Scalebio: Scalable bilevel optimization for llm data reweighting. *arXiv preprint arXiv:2406.19976*, 2024.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. Trak: Attributing model behavior at scale. *arXiv preprint arXiv:2303.14186*, 2023.

Jupinder Parmar, Sanjev Satheesh, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. Reuse, don't retrain: A recipe for continued pretraining of language models. *arXiv preprint arXiv:2407.07263*, 2024.

Amey Pasarkar and Adji Bousso Dieng. Cousins of the vendi score: A family of similarity-based diversity metrics for science and machine learning. *arXiv preprint arXiv:2310.12952*, 2023.

Keiran Paster, Marco Dos Santos, Zhangir Azerbayev, and Jimmy Ba. Openwebmath: An open dataset of high-quality mathematical web text. *arXiv preprint arXiv:2310.06786*, 2023.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are nlp models really able to solve simple math word problems? *arXiv preprint arXiv:2103.07191*, 2021.

Leena H Patil and Mohammed Atique. A novel approach for feature selection method tf-idf in document clustering. In *2013 3rd IEEE international advance computing conference (IACC)*, pp. 858–862. IEEE, 2013.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. *Advances in neural information processing systems*, 34:20596–20607, 2021.

Barak A. Pearlmutter. Fast exact multiplication by the hessian. *Neural Computation*, 6(1):147–160, 1994. doi: 10.1162/neco.1994.6.1.147.

Robert K Peet. The measurement of species diversity. *Annual review of ecology and systematics*, pp. 285–307, 1974.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. The refinedweb dataset for falcon llm: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*, 2023.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*, 2023.

Leif E Peterson. K-nearest neighbor. *Scholarpedia*, 4(2):1883, 2009.

Agustin Picard, Lucas Hervier, Thomas Fel, and David Vigouroux. Influenciæ: A library for tracing the influence back to the data-points. In *World Conference on Explainable Artificial Intelligence*, pp. 193–204. Springer, 2024.

Maria Priestley, Fionntán O'donnell, and Elena Simperl. A survey of data quality requirements that matter in ml development pipelines. *ACM Journal of Data and Information Quality*, 15(2):1–39, 2023.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. Estimating training data influence by tracing gradient descent. *Advances in Neural Information Processing Systems*, 33:19920–19930, 2020.

Yulei Qin, Xingyu Chen, Yunhang Shen, Chaoyou Fu, Yun Gu, Ke Li, Xing Sun, and Rongrong Ji. Capro: webly supervised learning with cross-modality aligned prototypes. *Advances in Neural Information Processing Systems*, 36, 2024.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. Improving language understanding by generative pre-training. 2018.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*, 2021.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances in Neural Information Processing Systems*, 36, 2024.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

K Raghuveer et al. Legal documents clustering using latent dirichlet allocation. *IAES Int. J. Artif. Intell*, 2 (1):34–37, 2012.

Xingcheng Ran, Yue Xi, Yonggang Lu, Xiangwen Wang, and Zhenyu Lu. Comprehensive survey on hierarchical clustering algorithms and the recent developments. *Artificial Intelligence Review*, 56(8):8219–8264, 2023.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

Alfréd Rényi. On measures of entropy and information. In *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability, volume 1: contributions to the theory of statistics*, volume 4, pp. 547–562. University of California Press, 1961.

Hamed Rezazadegan Tavakoli, Esa Rahtu, and Janne Heikkilä. Fast and efficient saliency detection using sparse sampling and kernel density estimation. In *Image Analysis: 17th Scandinavian Conference, SCIA 2011, Ystad, Sweden, May 2011. Proceedings 17*, pp. 666–675. Springer, 2011.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. Beyond accuracy: Behavioral testing of nlp models with checklist. *arXiv preprint arXiv:2005.04118*, 2020.

Brian Richards. Type/token ratios: What do they really tell us? *Journal of child language*, 14(2):201–209, 1987.

Yuji Roh, Geon Heo, and Steven Euijong Whang. A survey on data collection for machine learning: a big data-ai integration perspective. *IEEE Transactions on Knowledge and Data Engineering*, 33(4):1328–1347, 2019.

Rajendra Kumar Roul, Omanwar Rohit Devanand, and Sanjay Kumar Sahay. Web document clustering and ranking using tf-idf based apriori approach. *arXiv preprint arXiv:1406.5617*, 2014.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*, 2021.

Gayathri Saranathan, Mahammad Parwez Alam, James Lim, Suparna Bhattacharya, Soon Yee Wong, Martin Foltin, and Cong Xu. Dele: Data efficient llm evaluation. In *ICLR 2024 Workshop on Navigating and Addressing Data Problems for Foundation Models*.

Nikunj Saunshi, Arushi Gupta, Mark Braverman, and Sanjeev Arora. Understanding influence functions and datamodels via harmonic analysis. In *The Eleventh International Conference on Learning Representations*, 2022.

Timo Schick and Hinrich Schütze. It's not just size that matters: Small language models are also few-shot learners. *arXiv preprint arXiv:2009.07118*, 2020.

Andrea Schioppa, Polina Zablotskaia, David Vilar, and Artem Sokolov. Scaling up influence functions, 2021. URL `https://arxiv.org/abs/2112.03052`.

Stephanie Schoch, Ritwick Mishra, and Yangfeng Ji. Data selection for fine-tuning large language models using transferred shapley values. *arXiv preprint arXiv:2306.10165*, 2023.

Sarah E Schwarm and Mari Ostendorf. Reading level assessment using support vector machines and statistical language models. In *Proceedings of the 43rd annual meeting of the Association for Computational Linguistics (ACL'05)*, pp. 523–530, 2005.

Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

Burr Settles. Active learning literature survey. *Science*, 10(3):237–304, 1995.

Burr Settles. From theories to queries: Active learning in practice. In *Active learning and experimental design workshop in conjunction with AISTATS 2010*, pp. 1–18. JMLR Workshop and Conference Proceedings, 2011.

Claude Elwood Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27 (3):379–423, 1948.

Claude Elwood Shannon. A mathematical theory of communication. *ACM SIGMOBILE mobile computing and communications review*, 5(1):3–55, 2001.

Yunfan Shao, Linyang Li, Zhaoye Fei, Hang Yan, Dahua Lin, and Xipeng Qiu. Balanced data sampling for language model training with clustering. *arXiv preprint arXiv:2402.14526*, 2024.

Bin Shen and Luo Si. Non-negative matrix factorization clustering on multiple manifolds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 24, pp. 575–580, 2010.

Jonathan Richard Shewchuk et al. An introduction to the conjugate gradient method without the agonizing pain. 1994.

Zhengxiang Shi and Aldo Lipani. Don't stop pretraining? make prompt-based fine-tuning powerful learner. *Advances in Neural Information Processing Systems*, 36:5827–5849, 2023.

Zhengyan Shi, Adam X Yang, Bin Wu, Laurence Aitchison, Emine Yilmaz, and Aldo Lipani. Instruction tuning with loss over instructions. *arXiv preprint arXiv:2405.14394*, 2024.

Manli Shu, Jiongxiao Wang, Chen Zhu, Jonas Geiping, Chaowei Xiao, and Tom Goldstein. On the exploitability of instruction tuning. *Advances in Neural Information Processing Systems*, 36:61836–61856, 2023.

Raphael Shu, Hideki Nakayama, and Kyunghyun Cho. Generating diverse translations with sentence codes. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 1823–1827, 2019.

Luo Si and Jamie Callan. A statistical model for scientific readability. In *Proceedings of the tenth international conference on Information and knowledge management*, pp. 574–576, 2001.

Aditya Siddhant and Zachary C Lipton. Deep bayesian active learning for natural language processing: Results of a large-scale empirical study. *arXiv preprint arXiv:1808.05697*, 2018.

Fatimah Sidi, Payam Hassany Shariat Panahy, Lilly Suriani Affendey, Marzanah A Jabar, Hamidah Ibrahim, and Aida Mustapha. Data quality: A survey of data quality dimensions. In *2012 International Conference on Information Retrieval & Knowledge Management*, pp. 300–304. IEEE, 2012.

Stacy Silverman and Nan Bernstein Ratner. Measuring lexical diversity in children who stutter: Application of vocd. *Journal of fluency disorders*, 27(4):289–304, 2002.

Edward H Simpson. Measurement of diversity. *nature*, 163(4148):688–688, 1949.

Kristina P Sinaga and Miin-Shen Yang. Unsupervised k-means clustering algorithm. *IEEE access*, 8: 80716–80727, 2020.

Ankur Sinha, Pekka Malo, and Kalyanmoy Deb. A review on bilevel optimization: From classical to evolutionary approaches and applications. *IEEE transactions on evolutionary computation*, 22(2):276–295, 2017.

Samarth Sinha, Han Zhang, Anirudh Goyal, Yoshua Bengio, Hugo Larochelle, and Augustus Odena. Small-gan: Speeding up gan training using core-sets. In *International Conference on Machine Learning*, pp. 9005–9015. PMLR, 2020.

Eric Michael Smith, Melissa Hall, Melanie Kambadur, Eleonora Presani, and Adina Williams. " i'm sorry to hear that": Finding new biases in language models with a holistic descriptor dataset. *arXiv preprint arXiv:2205.09209*, 2022.

John Smith and Lisa Johnson. Strategies for difficulty sampling providing diversity in datasets. *Journal of Machine Learning Research*, 10:100–120, 2020.

Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning from noisy labels with deep neural networks: A survey. *IEEE transactions on neural networks and learning systems*, 34(11): 8135–8153, 2022.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. *Advances in Neural Information Processing Systems*, 35: 19523–19536, 2022.

Karen Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21, 1972.

Eleftherios Spyromitros-Xioufis, Symeon Papadopoulos, Alexandru Lucian Ginsca, Adrian Popescu, Yiannis Kompatsiaris, and Ioannis Vlahavas. Improving diversity in image search via supervised relevance scoring. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, pp. 323–330, 2015.

Katherine Stasaski and Marti A Hearst. Semantic diversity in dialogue with natural language inference. *arXiv preprint arXiv:2205.01497*, 2022.

Katherine Stasaski, Grace Hui Yang, and Marti A Hearst. More diverse dialogue datasets via diversity-informed data collection. In *Proceedings of the 58th annual meeting of the association for computational linguistics*, pp. 4958–4968, 2020.

Albert Yu Sun, Eliott Zemour, Arushi Saxena, Udith Vaidyanathan, Eric Lin, Christian Lau, and Vaikkunth Mugunthan. Does fine-tuning gpt-3 with the openai api leak personally-identifiable information? *arXiv preprint arXiv:2307.16382*, 2023.

Peng Sun, Bei Shi, Daiwei Yu, and Tao Lin. On the diversity and realism of distilled dataset: An efficient dataset distillation paradigm. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9390–9399, 2024a.

Wangtao Sun, Haotian Xu, Xuanqing Yu, Pei Chen, Shizhu He, Jun Zhao, and Kang Liu. Itd: Large language models can teach themselves induction through deduction. *arXiv preprint arXiv:2403.05789*, 2024b.

Jun Suzuki, Heiga Zen, and Hideto Kazawa. Extracting representative subset from extensive text data for training pre-trained language models. *Information Processing & Management*, 60(3):103249, 2023.

Luis Talavera. Feature selection as a preprocessing step for hierarchical clustering. In *ICML*, volume 99, pp. 389–397, 1999.

Haoru Tan, Sitong Wu, Fei Du, Yukang Chen, Zhibin Wang, Fan Wang, and Xiaojuan Qi. Data pruning via moving-one-sample-out. *Advances in Neural Information Processing Systems*, 36, 2024a.

Jiejun Tan, Zhicheng Dou, Yutao Zhu, Peidong Guo, Kun Fang, and Ji-Rong Wen. Small models, big insights: Leveraging slim proxy models to decide when and what to retrieve for llms. *arXiv preprint arXiv:2402.12052*, 2024b.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html*, 3(6):7, 2023.

MosaicML NLP Team. Introducing mpt-7b: A new standard for open-source, commercially usable llms, 2023. URL `www.mosaicml.com/blog/mpt-7b`. Accessed: 2023-05-05.

Mildred C Templin. *Certain language skills in children; their development and interrelationships.* University of Minnesota Press, 1957.

Guy Tevet and Jonathan Berant. Evaluating the evaluation of diversity in natural language generation. *arXiv preprint arXiv:2004.02990*, 2020.

Steve Tingiris and Bret Kinsella. *Exploring GPT-3: An unofficial first look at the general-purpose language processing API from OpenAI.* Packt Publishing Ltd, 2021.

Kushal Tirumala, Aram Markosyan, Luke Zettlemoyer, and Armen Aghajanyan. Memorization without overfitting: Analyzing the training dynamics of large language models. *Advances in Neural Information Processing Systems*, 35:38274–38290, 2022.

Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari Morcos. D4: Improving llm pretraining via document de-duplication and diversification. *Advances in Neural Information Processing Systems*, 36, 2024.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J Gordon. An empirical study of example forgetting during deep neural network learning. *arXiv preprint arXiv:1812.05159*, 2018.

Simon Tong and Daphne Koller. Support vector machine active learning with applications to text classification. *Journal of machine learning research*, 2(Nov):45–66, 2001.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023a.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023b.

Karina Vidal and Scott Jarvis. Effects of english-medium instruction on spanish students' proficiency and lexical diversity in english. *Language Teaching Research*, 24(5):568–587, 2020.

Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17:395–416, 2007.

Chi Wang, Qingyun Wu, Silu Huang, and Amin Saied. Economic hyperparameter optimization with blended search strategy. In *International Conference on Learning Representations*, 2021a.

Chi Wang, Qingyun Wu, Markus Weimer, and Erkang Zhu. Flaml: A fast and lightweight automl library. *Proceedings of Machine Learning and Systems*, 3:434–447, 2021b.

Jiahao Wang, Bolin Zhang, Qianlong Du, Jiajun Zhang, and Dianhui Chu. A survey on data selection for llm instruction tuning. *arXiv preprint arXiv:2402.05123*, 2024a.

Lingzhi Wang, Xingshan Zeng, Jinsong Guo, Kam-Fai Wong, and Georg Gottlob. Selective forgetting: Advancing machine unlearning techniques and evaluation in language models. *arXiv preprint arXiv:2402.05813*, 2024b.

Yidong Wang, Zhuohao Yu, Zhengran Zeng, Linyi Yang, Cunxiang Wang, Hao Chen, Chaoya Jiang, Rui Xie, Jindong Wang, Xing Xie, et al. Pandalm: An automatic evaluation benchmark for llm instruction tuning optimization. *arXiv preprint arXiv:2306.05087*, 2023a.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. Self-instruct: Aligning language models with self-generated instructions. *arXiv preprint arXiv:2212.10560*, 2022.

Yu-Xiong Wang and Yu-Jin Zhang. Nonnegative matrix factorization: A comprehensive review. *IEEE Transactions on knowledge and data engineering*, 25(6):1336–1353, 2012.

Yufei Wang, Wanjun Zhong, Liangyou Li, Fei Mi, Xingshan Zeng, Wenyong Huang, Lifeng Shang, Xin Jiang, and Qun Liu. Aligning large language models with human: A survey. *arXiv preprint arXiv:2307.12966*, 2023b.

Yulin Wang, Gao Huang, Shiji Song, Xuran Pan, Yitong Xia, and Cheng Wu. Regularizing deep networks with semantic data augmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44 (7):3733–3748, 2021c.

Kellie Webster, Xuezhi Wang, Ian Tenney, Alex Beutel, Emily Pitler, Ellie Pavlick, Jilin Chen, Ed Chi, and Slav Petrov. Measuring and reducing gendered correlations in pre-trained models. *arXiv preprint arXiv:2010.06032*, 2020.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

Max Welling. Herding dynamical weights to learn. In *Proceedings of the 26th annual international conference on machine learning*, pp. 1121–1128, 2009.

Alexander Wettig, Aatmik Gupta, Saumya Malik, and Danqi Chen. Qurating: Selecting high-quality data for training language models. *arXiv preprint arXiv:2402.09739*, 2024.

Svante Wold, Kim Esbensen, and Paul Geladi. Principal component analysis. *Chemometrics and intelligent laboratory systems*, 2(1-3):37–52, 1987.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*, 2019.

Haolun Wu, Yansen Zhang, Chen Ma, Fuyuan Lyu, Fernando Diaz, and Xue Liu. A survey of diversification techniques in search and recommendation. *CoRR arXiv*, 2212, 2022.

Haolun Wu, Yansen Zhang, Chen Ma, Fuyuan Lyu, Bowei He, Bhaskar Mitra, and Xue Liu. Result diversification in search and recommendation: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 2024.

Shengguang Wu, Keming Lu, Benfeng Xu, Junyang Lin, Qi Su, and Chang Zhou. Self-evolved diverse data sampling for efficient instruction tuning. *arXiv preprint arXiv:2311.08182*, 2023.

Mengzhou Xia, Tianyu Gao, Zhiyuan Zeng, and Danqi Chen. Sheared llama: Accelerating language model pre-training via structured pruning. *arXiv preprint arXiv:2310.06694*, 2023.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*, 2024a.

Xiaobo Xia, Jiale Liu, Jun Yu, Xu Shen, Bo Han, and Tongliang Liu. Moderate coreset: A universal method of data selection for real-world data-efficient deep learning. In *The Eleventh International Conference on Learning Representations*, 2022.

Xiaobo Xia, Jiale Liu, Shaokun Zhang, Qingyun Wu, Hongxin Wei, and Tongliang Liu. Refined coreset selection: Towards minimal coreset size under model performance constraints. In *Forty-first International Conference on Machine Learning*, 2024b.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. Unsupervised data augmentation for consistency training. *Advances in neural information processing systems*, 33:6256–6268, 2020.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*, 2021.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. Data selection for language models via importance resampling. *Advances in Neural Information Processing Systems*, 36:34201–34227, 2023.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, and Daxin Jiang. Wizardlm: Empowering large language models to follow complex instructions. *arXiv preprint arXiv:2304.12244*, 2023a.

Frank F Xu, Uri Alon, Graham Neubig, and Vincent Josua Hellendoorn. A systematic evaluation of large language models of code. In *Proceedings of the 6th ACM SIGPLAN International Symposium on Machine Programming*, pp. 1–10, 2022.

Yang Xu, Yongqiang Yao, Yufan Huang, Mengnan Qi, Maoquan Wang, Bin Gu, and Neel Sundaresan. Rethinking the instruction quality: Lift is what you need. *arXiv preprint arXiv:2312.11508*, 2023b.

Zhangchen Xu, Fengqing Jiang, Luyao Niu, Yuntian Deng, Radha Poovendran, Yejin Choi, and Bill Yuchen Lin. Magpie: Alignment data synthesis from scratch by prompting aligned llms with nothing. *arXiv preprint arXiv:2406.08464*, 2024.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

Yaoqing Yang, Ryan Theisen, Liam Hodgkinson, Joseph E Gonzalez, Kannan Ramchandran, Charles H Martin, and Michael W Mahoney. Evaluating natural language processing models with generalization metrics that do not need access to any training or testing data. *arXiv preprint arXiv:2202.02842*, 2022.

Emmanuel J Yannakoudakis and David Fawthrop. The rules of spelling errors. *Information Processing & Management*, 19(2):87–99, 1983.

Gregory Yauney, Emily Reif, and David Mimno. Data similarity is not enough to explain language model performance. *arXiv preprint arXiv:2311.09006*, 2023.

Jingwen Ye, Ruonan Yu, Songhua Liu, and Xinchao Wang. Distilled datamodel with reverse gradient matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11954–11963, 2024.

Çağatay Yıldız, Nishaanth Kanna Ravichandran, Prishruit Punia, Matthias Bethge, and Beyza Ermis. Investigating continual pretraining in large language models: Insights and implications. *arXiv preprint arXiv:2402.17400*, 2024.

Julio Christian Young and Makoto Shishido. Investigating openai's chatgpt potentials in generating chatbot's dialogue for english as a foreign language learning. *International journal of advanced computer science and applications*, 14(6), 2023.

Zichun Yu, Spandan Das, and Chenyan Xiong. Mates: Model-aware data selection for efficient pretraining with data influence models. *arXiv preprint arXiv:2406.06046*, 2024.

Beverly L Zakaluk and S Jay Samuels. *Readability: Its Past, Present, and Future.* ERIC, 1988.

Zhiyuan Zeng, Jiatong Yu, Tianyu Gao, Yu Meng, Tanya Goyal, and Danqi Chen. Evaluating large language models at evaluating instruction following. *arXiv preprint arXiv:2310.07641*, 2023.

Daochen Zha, Zaid Pervaiz Bhat, Kwei-Herng Lai, Fan Yang, Zhimeng Jiang, Shaochen Zhong, and Xia Hu. Data-centric artificial intelligence: A survey. *arXiv preprint arXiv:2303.10158*, 2023.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning (still) requires rethinking generalization. *Communications of the ACM*, 64(3):107–115, 2021.

Chiyuan Zhang, Daphne Ippolito, Katherine Lee, Matthew Jagielski, Florian Tramèr, and Nicholas Carlini. Counterfactual memorization in neural language models. *Advances in Neural Information Processing Systems*, 36:39321–39362, 2023a.

Jipeng Zhang, Yaxuan Qin, Renjie Pi, Weizhong Zhang, Rui Pan, and Tong Zhang. Tagcos: Task-agnostic gradient clustered coreset selection for instruction tuning data. *arXiv preprint arXiv:2407.15235*, 2024a.

Jizhi Zhang, Keqin Bao, Yang Zhang, Wenjie Wang, Fuli Feng, and Xiangnan He. Is chatgpt fair for recommendation? evaluating fairness in large language model recommendation. In *Proceedings of the 17th ACM Conference on Recommender Systems*, pp. 993–999, 2023b.

Lei Zhang. Bilevel optimization in the deep learning era: Methods and applications. 2024.

Shaokun Zhang, Xiaobo Xia, Zhaoqing Wang, Ling-Hao Chen, Jiale Liu, Qingyun Wu, and Tongliang Liu. Ideal: Influence-driven selective annotations empower in-context learners in large language models. *arXiv preprint arXiv:2310.10873*, 2023c.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*, 2023d.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*, 2019.

Xiao Zhang and Ji Wu. Dissecting learning and forgetting in language model finetuning. In *The Twelfth International Conference on Learning Representations*, 2024.

Xiaoying Zhang, Baolin Peng, Ye Tian, Jingyan Zhou, Yipeng Zhang, Haitao Mi, and Helen Meng. Self-tuning: Instructing llms to effectively acquire new knowledge through self-teaching. *arXiv preprint arXiv:2406.06326*, 2024b.

Yifan Zhang, Yifan Luo, Yang Yuan, and Andrew C Yao. Autonomous data selection with language models for mathematical texts. In *ICLR 2024 Workshop on Navigating and Addressing Data Problems for Foundation Models*, 2024c.

Yihua Zhang, Yuguang Yao, Parikshit Ram, Pu Zhao, Tianlong Chen, Mingyi Hong, Yanzhi Wang, and Sijia Liu. Advancing model pruning via bi-level optimization. *Advances in Neural Information Processing Systems*, 35:18309–18326, 2022.

Bo Zhao and Hakan Bilen. Dataset condensation with distribution matching. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 6514–6523, 2023.

Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching. *arXiv preprint arXiv:2006.05929*, 2020a.

Bo Zhao, Konda Reddy Mopuri, and Hakan Bilen. Dataset condensation with gradient matching, 2021. URL `https://arxiv.org/abs/2006.05929`.

Chenyang Zhao, Xueying Jia, Vijay Viswanathan, Tongshuang Wu, and Graham Neubig. Self-guide: Better task-specific instruction following via self-synthetic finetuning. *arXiv preprint arXiv:2407.12874*, 2024a.

Dora Zhao, Jerone TA Andrews, Orestis Papakyriakopoulos, and Alice Xiang. Position: Measure dataset diversity, don't just claim it. *arXiv preprint arXiv:2407.08188*, 2024b.

Dorothy Zhao, Jerone TA Andrews, AI Sony, Tokyo Orestis Papakyriakopoulos, and Alice Xiang. Measuring diversity in datasets. *International Conference on Learning Representations*, 2024c.

Shanshan Zhao, Mingming Gong, Tongliang Liu, Huan Fu, and Dacheng Tao. Domain generalization via entropy regularization. *Advances in neural information processing systems*, 33:16096–16107, 2020b.

Shuaijiang Zhao and Xiaoquan Fang. Technical report: Competition solution for bettermixture. *arXiv preprint arXiv:2403.13233*, 2024.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36, 2024.

Ming Zhong, Yang Liu, Da Yin, Yuning Mao, Yizhu Jiao, Pengfei Liu, Chenguang Zhu, Heng Ji, and Jiawei Han. Towards a unified multi-dimensional evaluator for text generation. *arXiv preprint arXiv:2210.07197*, 2022.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36, 2024a.

Daquan Zhou, Kai Wang, Jianyang Gu, Xiangyu Peng, Dongze Lian, Yifan Zhang, Yang You, and Jiashi Feng. Dataset quantization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 17205–17216, 2023.

Jianghong Zhou, Eugene Agichtein, and Surya Kallumadi. Diversifying multi-aspect search results using simpson's diversity index. In *Proceedings of the 29th ACM International conference on information & knowledge management*, pp. 2345–2348, 2020.

Yuhan Zhou, Fengjiao Tu, Kewei Sha, Junhua Ding, and Haihua Chen. A survey on data quality dimensions and tools for machine learning. *arXiv preprint arXiv:2406.19614*, 2024b.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. Judgelm: Fine-tuned large language models are scalable judges. *arXiv preprint arXiv:2310.17631*, 2023.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. Texygen: A benchmarking platform for text generation models. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pp. 1097–1100, 2018.