Du F, Ma XJ, Yang JR *et al*. A survey of LLM datasets: From autoregressive model to AI chatbot. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 39(3): 542-566 May 2024. DOI: 10.1007/s11390-024-3767-3

A Survey of LLM Datasets: From Autoregressive Model to AI Chatbot

Fei Du[1, 2, 3] (杜　非), Xin-Jian Ma[1, 2, *] (马新建), *Member, CCF*, Jing-Ru Yang[1, 2] (杨婧如)
Yi Liu[1, 2] (柳　熠), *Member, CCF, IEEE*, Chao-Ran Luo[1, 2] (罗超然), Xue-Bin Wang[1, 2] (王学斌)
Hai-Ou Jiang[1, 2] (姜海鸥), and Xiang Jing[1, 2, 4, *] (景　翔), *Member, CCF*

[1] *National Key Laboratory of Data Space Technology and System, Beijing 100195, China*

[2] *Advanced Institute of Big Data, Beijing 100195, China*

[3] *Fu Foundation School of Engineering and Applied Science, Columbia University, NY 10027, U.S.A.*

[4] *School of Software and Microelectronics, Peking University, Beijing 100091, China*

E-mail: fd2547@columbia.edu; maxj@aibd.ac.cn; okiyang@aibd.ac.cn; liuyi315@aibd.ac.cn; luochaoran@aibd.ac.cn
wxb_nudt@aibd.ac.cn; jiangho@aibd.ac.cn; jingxiang@pku.edu.cn

Received September 13, 2023; accepted April 25, 2024.

Abstract　Since OpenAI opened access to ChatGPT, large language models (LLMs) become an increasingly popular topic attracting researchers' attention from abundant domains. However, public researchers meet some problems when developing LLMs given that most of the LLMs are produced by industries and the training details are typically unrevealed. Since datasets are an important setup of LLMs, this paper does a holistic survey on the training datasets used in both the pre-train and fine-tune processes. The paper first summarizes 16 pre-train datasets and 16 fine-tune datasets used in the state-of-the-art LLMs. Secondly, based on the properties of the pre-train and fine-tune processes, it comments on pre-train datasets from quality, quantity, and relation with models, and comments on fine-tune datasets from quality, quantity, and concerns. This study then critically figures out the problems and research trends that exist in current LLM datasets. The study helps public researchers train and investigate LLMs by visual cases and provides useful comments to the research community regarding data development. To the best of our knowledge, this paper is the first to summarize and discuss datasets used in both autoregressive and chat LLMs. The survey offers insights and suggestions to researchers and LLM developers as they build their models, and contributes to the LLM study by pointing out the existing problems of LLM studies from the perspective of data.

Keywords　large language model (LLM), autoregressive model, AI chatbot, natural language processing (NLP) corpora, OpenAI

1　Introduction

With the emergence of ChatGPT①, large language models (LLMs) become a key topic due to their outstanding performance on communication tasks[1]. Because of their superior abilities, many researchers want to build their own "ChatGPT" these days. However, this meets some problems since the training details including the training process and training datasets for modern LLMs remain unclear in their descriptions[2], as most of the LLMs are produced by industries for commercial or other practical reasons. This makes model reproduction and modeling research for public scholars difficult. Thus our motivation for composing this paper is to help researchers train and investigate LLMs by offering some insights and suggestions regarding training data, as well as pointing out the existing problems in LLM corpora. This paper focuses on dataset constructions in pre-trained LLMs with decoder-only Transformer[3] architecture (autoregressive models) and fine-tune LLMs further trained for dialogue or communication as they are the most concerned sub-topics. To the best of our knowledge, this paper is the first to summarize and

Survey

*Corresponding Author (Xin-Jian Ma is responsible for guiding the content of the paper; Xiang Jing is responsible for the design of the paper's overall structure.)

①https://openai.com/blog/chatgpt, May 2024.

discuss datasets used in both autoregressive and chat LLMs.

The importance of constructing appropriate training data is emphasized by the non-replicability of the pre-train process and the ability to influence the model directly in the fine-tune process. Inspired by the Scaling Law[4], which says that large language models perform better when they include more parameters and absorb more training data, recent LLMs require billions or trillions of tokens of data to finish their pre-train process[5]. The cost of computation is so high, making it impossible to repeat the training process[2]. Compared with pre-train, fine-tune is not expensive and easy to reproduce. Fine-tune data always focuses on a specific domain to enhance the targeted ability of pre-train LLMs. For example, researchers need human-instruction data to fine-tune dialogue models to make them more helpful and aligned with human preference[6].

LLMs are a broad topic and there are many surveys in this area currently. Some comprehensive surveys examine LLMs from data preparation to model application (e.g., [2, 7]). Most of the surveys focus on the downstream application of LLMs (e.g., [8–13]). As a comparison, this paper researches the step of data construction in LLM modeling. There is a similar idea in the study of Chang *et al.*, which reviews the step of evaluation[14]. Besides, it narrows the topic down to decoder-only and chat models to make our research comprehensive and deep in content. Current work in this domain often starts from a single model like ChatGPT[15–17] and GPT-3[18–20]. Few studies analyze the training method of conversational AI[21] in general. Compared with these studies, our work is not limited to a single model and focuses on data instead of the training details. Unlike many studies concerning the issues brought by applications of LLMs[22–26], this paper also critically figures out the problems and trends that exist in LLM datasets. This brings a new perspective for researchers to think about the current situation of LLM studies. Though decoder LLM and chat models are popular topics, there is no systematic review of LLM datasets in this domain so far.

The paper first introduces the background of LLM datasets including data usage and operation in Section 2. Then it presents a systematic review of corpora used in recent popular LLMs in detail, including data used in both the pre-train and fine-tune processes. The survey summarizes the datasets used in the two processes and then analyzes them from different perspectives to answer the following questions.

- What LLM datasets should people use and how should they use them? (Subsections 3.1.2, 3.1.3, 3.2.3, and Section 4)
- What insights do datasets used in current LLMs reveal? (Subsection 3.2.2 and Section 5)

The main contributions of this paper are:

- summarizing 16 pre-train datasets and 16 fine-tune datasets for LLM researchers and developers to construct training data conveniently;
- providing comments about choosing or creating pre-train and fine-tune datasets by examining the quality, quantity, and function of data;
- figuring out the constraint of development for public LLMs and other problems associated with LLM corpora to offer insights to researchers.

2 Background

OpenAI first used the term "large language model" in GPT-2[27] of its GPT model series, which has 1.5 billion parameters. At the same time, the popular language model, Bert[28], has only 340 million parameters. Both models are based on the architecture of Transformer[3], which can be divided into the encoder and decoder parts. The difference is that the GPT series uses the decoder-only architecture derived from Transformer, whereas Bert uses a bidirectional encoder architecture. Encoder and decoder are not the only two derivations of the original Transformer. Some other language models like GLM[29] also use encoder-decoder transformers. The concept of LLM became well-known when OpenAI introduced ChatGPT, which is fine-tuned on the base model GPT-3[30] using the technique called instruction tuning and reinforcement learning from human feedback (RLHF). These methods are first introduced by InstructGPT[31], which is a milestone of chatting LLMs[16].

In recent years, a normal procedure of training LLMs is pre-train and fine-tune[32, 33], which is illustrated in Fig.1. The pre-train process ensures that the model can be used in a bunch of downstream tasks, and the fine-tune process enhances a specific aspect of the model, such as conversation[34]. This procedure is widely adopted because training large-size models in one step is resource-consuming[35], but the fine-tune process is efficient and affordable. Thus splitting the process to pre-train and fine-tune allows the model to economically perform more tasks.

In this section, the paper briefly describes how the

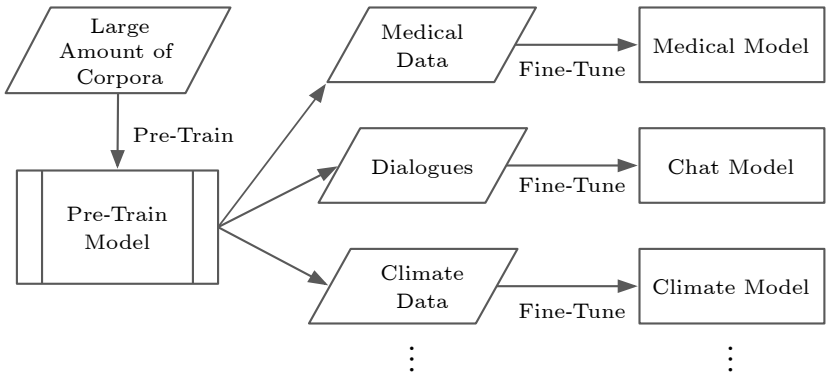


Fig.1. Training procedures: pre-train and fine-tune.

data is gathered and used in the pre-train process and the fine-tune process.

2.1 Pre-Train

The pre-train data is constructed by combining datasets from different sources and is generally operated through three steps, i.e., filtering, deduplication, and adding weights (optional), to increase the overall data quality.

Filtering. There are two categories of filtering, quality filtering and language filtering. The former is always required and the latter depends on the team's requirement for the model (e.g., BLOOM[36] is a multilingual LLM thus it must include different languages). Quality filtering is essential since raw data contains some unwanted speeches like abusive language[37]. There are two normal choices for quality filtering, researchers can either train a binary classifier to recognize high-quality data (e.g., [30, 38]) or set some thresholds (like the length of a sentence) to pick high-quality data manually (e.g., [39, 40]). Both methods are widely used in recent studies.

Deduplication. Deduplication is important because this process benefits models by improving model perplexity[41]. Deduplication has various levels, and a common way is to deduplicate at the document level, which is removing the document containing similar content when compared with one or more existing documents. An easy way to deduplicate is that researchers can remove the document when they find that the contents exactly match. However, this is not common in reality. Another way is to use the n-grams method. This is a widely used deduplication strategy in LLM training (e.g., [30]). This method allows people to find the number of overlapped phrases in a document, and then researchers decide whether to remove the document or the phrases by using thresholds.

Adding Weights. Although there is a huge amount of NLP (natural language processing) data from the Internet, the amount of high-quality data occupies only a small proportion. Since high-quality data is better for pre-train[42, 43], researchers might want to set higher weights to data with high-quality and lower weights to raw or low-quality data. One strategy is to offer different epochs elapsed to different subsets. In the pre-train process of GPT-3, the OpenAI team set the epochs elapsed to be 3.4 for Wikipedia and 0.44 for Common Crawl (after filtering)[30]. Repeating is a common method to deal with the problem of lacking high-quality data, but this might also be harmful to the model. It might let the model overfit on the data repeated many times[30]. A current study about multi-epoch repetition states that repeated tokens will cause model performance degradation, especially for large-size models[5].

2.2 Fine-Tune

Fine-tune datasets are constructed along with different models, and they do not require complicated operations after construction. In the fine-tune processes of chat models, data is used to let models align with human values, which are quality, truthfulness, and safety[31, 32].

Quality. Quality represents both human common sense and the subjectivity and specificity of outputs[44]. It should not only answer the user's question but also ideally narrate it interestingly. LaMDA-FT uses an explicit method to train the quality of the model as it uses binary labels representing sensibleness, specificity, and interestingness[32]. Helpfulness or quality is the benchmark that is the easiest to tune and can even outperform real human beings (e.g., [32, 45]).

Truthfulness. A big problem fine-tune LLMs are facing is hallucinations[31, 33]. Models tend to generate

plausible output when they do not have the correct answer. A recent study suggested the problem of hallucination originated in the pre-train process of model training due to memorization and corpus-based heuristics[46]. The fine-tune process can improve the truthfulness by evaluating the result of the model output, but results also show that the truthfulness of LLMs after fine-tuning is hard to be competitive compared with human beings[31, 32].

Safety. Another problem of LLMs is toxicity. As LLMs become practical tools in daily life, it is important to ensure that the outputs are harmless. A recent fine-tune process using a human inference gives only a small improvement in toxicity[31]. The shortcoming of human inference techniques is also reflected by the inefficiency in reducing the toxicity of ChatGPT when given persona (using prompts to let the model speak like a specific person on the system level)[47].

3 Systematic Review of Datasets

In this section, the paper reviews some primary datasets used in pre-train and fine-tune processes.

For pre-train datasets, the paper divides the corpora into five categories based on the types of data resources that are crucial to decoder pre-train models and also their functions in the future fine-tune process; the five categories are web text (the large amount of corpora crawled from web pages offer a general understanding of language), book (old books without offending the copy right as long consecutive text benefit the long-text understanding), academy (journals or other scientific text can improve the cleanness of output), dialogue (dialogues with human intervention enhance models' performance on chatting), and code (codes from sources like GitHub are beneficial to model both for logistic and downstream coding skills). The detailed functions for the datasets are discussed in Subsection 3.1.2.

For each class of data the paper analyzes several datasets as well as the decoder model that utilizes them by examining Xavier's summary[48]. This paper ensures that all the datasets have been used in state-of-the-art models so that the functions and usage are justified to a certain degree.

Since there are demands for non-English corpora, this paper also lists some large-size datasets with other languages at the end of Subsection 3.1.1. The paper uses brief descriptions since they do not reveal special insights (other than helping models generate different languages) and are not included in the models this paper focuses on.

Similarly, for fine-tune datasets, this paper divides the datasets into three categories: comparison data, instruction tuning data, and conversation data. Detailed contributions of these classes of data are discussed in Subsection 3.2.3. Then the paper uses the same procedure as before; instead the focused models are fine-tune chatbots. It also includes some fine-tune datasets that are not used in decoder-based fine-tune models but are insightful in research.

The final list of models includes GPT-3[30], pre-trained LaMDA (LaMBDA-PT)[32], LLaMA[39], PaLM[49], Gopher[40], and Chinchilla[50] for pre-train models, and fine-tune LaMDA (LaMBDA-FT)[32], InstructGPT[31], Alpaca②, Vicuna[34], GPT Self-Inst[51], ChatGPT, GPT-4[33], and LLaMA-GPT4[45] for fine-tuned models.

3.1 Pre-Train Datasets

Pre-train data is composed of datasets from different sources, and each dataset has its own property and serves a certain function. Fig.2 shows pre-train datasets in different categories. Table 1 summarizes all the datasets and provides links for easy access. The paper emphasizes open-sourced datasets, and it includes two private datasets because other open-sourced datasets in their categories are their reproductions.

3.1.1 Datasets for Pre-Train Models

This paper classifies the data used in the pre-train model into five categories: web text, book, academy, dialogue, and code. Though data is important for developing downstream tasks, most pre-train models will not use data from all classes. Pre-train models often use data from three to four categories. The paper would like to present those datasets in two parts, open-sourced datasets and private datasets.

Open-sourced datasets offer download links or codes for data operation that allow researchers to download the data directly or reproduce the data by themselves. They are most helpful to people who want to build their models. Currently, there are some researchers working on building public-accessed datasets to contribute to the research of LLMs (e.g., [52, 53]), and Peking University is working on connecting

②https://github.com/tatsu-lab/stanford_alpaca, May 2024.

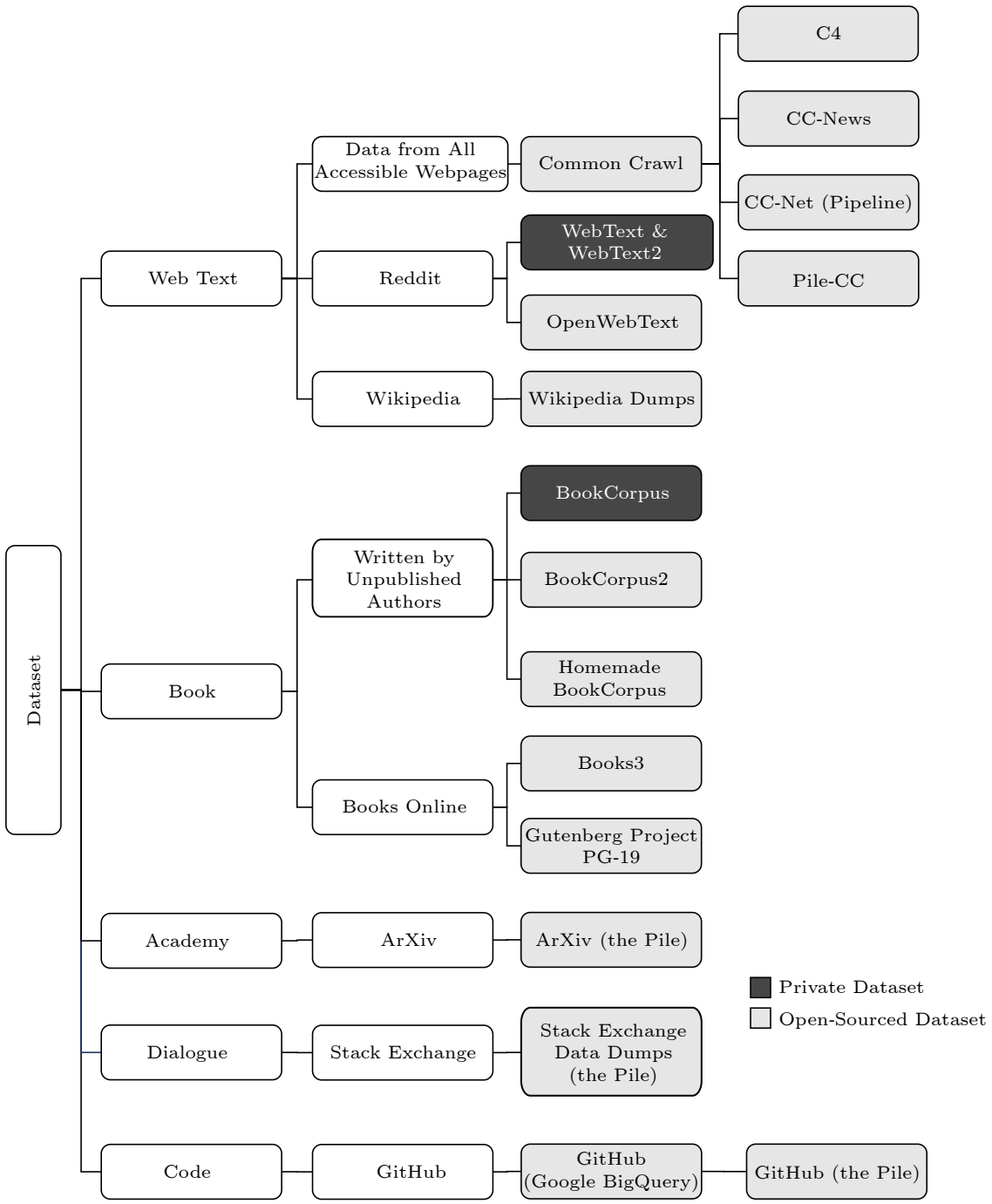


Fig.2. Categories of pre-train datasets.

different data sources to make the data easily accessible to the public[54]. The open-sourced datasets this paper will introduce are Common Crawl③, C4[55], English Wikipedia, OpenWebText (1 and 2)[52], GitHub, Homemade BookCorpus, Gutenberg Project PG-19[56], The Pile[52], BookCorpus2[52], Books3[52], and CC-Net[57].

Common Crawl. Common Crawl is a big open-source project. It crawls data from the Internet without filtering, and the data updates every month. This is one of the biggest open-source corpus datasets on the Internet, containing lots of raw data. There are many versions of Common Crawl created by different teams after filtering the raw data downloaded from the Common Crawl website③. Common Crawl as a

③https://commoncrawl.org/, May 2024.

Table 1. Summary of Pre-Train Datasets

Dataset	Size	Language	Brief Introduction	Link
Common Crawl	Updates every month	(No statistic)	Raw webpage data	https://commoncrawl.org/the-data/get-started/
C4	750 GB	English	Common Crawl data after filtering	https://github.com/google-research/text-to-text-transfer-transformer#datasets
CC-Net	3.2 TB	English ≈ 61.09%, Russian ≈ 10.93%	Common Crawl data after filtering	https://github.com/facebookresearch/cc_net
English Wikipedia	6 GB (20200301)	English > 93%	Wikipedia data	https://dumps.wikimedia.org/
GitHub (Google BigQuery)	3 TB+	(Code)	Code and texts from GitHub	https://console.cloud.google.com/bigquery?p=bigquery-public-data&d=github_repos
The Pile	825 GB	English ≈ 97.4%	Mixed data	
OpenWebText (1 and 2)	38 GB and 63 GB	English	Webpage data from Reddit	https://skylion007.github.io/OpenWebTextCorpus/, https://github.com/EleutherAI/the-pile
BookCorpus2	6 GB	(No statistic)	Online books	https://github.com/EleutherAI/the-pile
Books3	101 GB	(No statistic)	Collected based on Bibliotik	https://github.com/EleutherAI/the-pile
Gutenberg Project (PG-19)	11 GB	(No statistic)	Books before 1919	https://github.com/deepmind/pg19
WebText (1 and 2)	40 GB and 96 GB	English	Webpage data from Reddit	Private data

hub of webpage data is powerful in both diversity and quantity. The project itself can offer five trillion tokens, which is sufficient to train large models with hundreds of billions of tokens[58]. As discovered by Alycia L. *et al.*[59], data crawled from webpages is also more diverse compared with data like Wikipedia.

C4 (Colossal Clean Crawled Corpus). C4[55] is an unlabeled dataset based on Common Crawl. It sets several criteria (such as there must be five or more sentences in a webpage) to filter sentences and webpages. C4 is a dataset after deduplication and the team uses Langdetect to filter out all the non-English webpages. C4 is a popular derivation of Common Crawl and is widely used in the pre-train process of many LLMs (e.g., LaMDA[32], Gopher[40]). It cannot be downloaded directly, but people can construct the dataset using the open-accessed tool.

English Wikipedia or Wikipedia. English Wikipedia or Wikipedia is mentioned in many LLM papers, but many papers do not have a clear description of what they refer to (e.g., GPT-3[30], LaMDA[32], LLaMA[39], and Gopher[40]). One way is to download data from Wiki dumps, which contain about 20 types of languages. The English dataset crawled from en.wikipedia.org has been used as evidence corpus[60]. Since Wikipedia is a relatively professional platform, LLM researchers normally consider this data as a high-quality corpus.

OpenWebText and OpenWebText2. OpenWebText④ is a dataset created using the pipeline described by the OpenAI team in creating the WebText dataset. The team crawls hyperlinks from the Reddit submissions dataset and uses FastText to filter out all non-English webpages. This dataset is deduplicated using local-sensitivity hashing (5-grams). OpenWebText2[52] extends the content of OpenwebText.

GitHub (Google BigQuery). The Google company cooperates with GitHub and provides its data through BigQuery in the Google Cloud platform. As suggested by the LLaMA team, this dataset can be filtered by only the remaining GitHub pages containing Apache, BSD, and MIT licenses. Another possible operation is to filter out uncommon programming languages, as suggested by the PaLM team. This is a raw dataset and researchers need to deduplicate before using it.

Homemade BookCorpus. Homemade BookCorpus⑤ is created by a project set up in GitHub using the same procedure as creating the original version of BookCorpus. Since the original version of BookCorpus is not open to the public in any way now, this dataset becomes one of the substitutes. It contains data crawls from smashwords.com and is formatted as one sentence per line.

Gutenberg Project PG-19. Gutenberg Project is an

④http://Skylion007.github.io/OpenWebTextCorpus, May 2024.
⑤https://github.com/soskek/bookcorpus, May 2024.

open-source webpage[⑥] that contains many electrical books. The PG-19 dataset[56] contains books from the Gutenberg Project. These books were published before 1919, thus there is no violation regarding the copyrights. This dataset also substitutes all the discriminative words with a special token.

The Pile. The Pile[52] is a combined dataset containing 22 subsets. Researchers can use the whole dataset or use only the subset they want. It contains abundant data including web page, academic knowledge, code, dialogue, books, email, social media, and so on. Some main subsets include PileCC, Books3, ArXiv, GitHub, and Stack Exchange. The Pile has been deduplicated at the document level.

BookCorpus2. BookCorpus2[52] is created using a similar pipeline as Homemade BookCorpus. The main modifications it makes include the chapter structure, the tables, and the code structure, so that the source code is more coherent.

Books3. Books3 is a subset of the Pile[52] and is used in LLaMA pre-train. It is based on Bibliotik which contains fiction and non-fiction books. This dataset is much larger than BookCorpus2 and Gutenberg Project PG-19. It contains about 100 GB data whereas the other two only contain about 10 GB data.

CC-Net. CC-Net[57] is a pipeline used to filter raw Common Crawl data. The steps are deduplication, language recognition (use FastText to detect the language and remove all pages which have unclear language), and filtering (calculate the degree of similarity between the data with high-quality corpus like Wikipedia and filter out low-quality data). This pipeline is used in the LLaMA pre-train.

Due to various reasons like commercial usage, many training datasets are private. Here the paper provides concise descriptions of how these datasets are created to offer some insights to researchers about dataset building. The private datasets include Webtext (1 and 2)[27, 30], Common Crawl (GPT-3)[30], and MassiveWeb[40, 50].

WebText and WebText2. WebText series[27, 30] is created by the OpenAI team and is used in both GPT-2 and GPT-3 pre-train. The OpenAI team points out that Common Crawl has low quality regarding the context, and creates WebText by crawling all the hyperlinks from Reddit posts having karma higher than 3. This dataset is after deduplication.

Common Crawl (GPT-3 Version). This dataset is also created by the OpenAI team and is used in GPT-3 pre-train. The OpenAI team filters the Common Crawl dataset using a logistic regression model built by comparing high-quality corpus like WebText with the raw Common Crawl data. This dataset is deduplicated.

MassiveWeb. MassiveWeb[40, 50] is a private webpage dataset produced by the DeepMind team. It is used in the pre-train of Gopher and Chinchilla models. The DeepMind team uses a self-created HTML crawler to collect only text data from web pages. This dataset is manipulated in six steps: content filtering (filtering out all non-English text), test extraction, quality filtering (according to the length of words, the characters, and the number or proportion of words in a line or page), repetition removal (removing documents with excessive repeated words or phrases), document deduplication, test-set filtering (ensuring that there is no overlap between the training set and testing set).

The lack of non-English corpora is a significant problem in today's LLM research (detailed statistics in Section 5), thus this paper briefly discusses some open-sourced large-scaled non-English monolingual datasets to help public researchers to pre-train models in other languages. The paper selects these non-English monolingual corpora based on the research on model BLOOM[36], which is the autoregressive LLM trained with the most amount of non-English text to the best of our knowledge.

WuDao. WuDao[61] is a large-scale Chinese corpus with 3 TB training data and 1.08 TB Chinese characters. The dataset is after cleaning and personal data has been removed.

Arabic Billion Words. Arabic Billion Words[62] is a 10 GB corpus containing over 1.5 billion Arabic words. It is collected from different countries speaking Arabic. Researchers need to clean the data before usage as the original producers did not mention the data cleaning step.

Indic NLP Corpus. Indic NLP Corpus[63] contains 2.7 billion words with 10 Indian languages and it has been shown that the resources developed on this dataset perform well on many NLP tasks.

Catalan Textual Corpus. Catalan Textual Corpus[64] is a clean Catalan dataset with 11 GB corpus and contains over 1.7 billion tokens.

[⑥] https://www.gutenberg.org, May 2024.

Normally, models use data from scientific sources, books, and web pages. Comments regarding the choice of datasets based on functions are provided in Subsection 3.1.2.

3.1.2 Functions of Datasets

In this subsection, the paper presents the relation between datasets and pre-train models in general (shown in Table 2), and provides some comments regarding the function of each class of datasets by summarizing dataset utilizations. LLM researchers and developers can choose datasets that fit the purpose of their models based on this summary. Recently, LLMs are expected to perform well on general natural language tasks, as well as domain-specific ones[14]. Thus this paper presents the functions of datasets from two aspects: enhancing a model's performance on language tasks and on specific downstream tasks. For each part, the paper also suggests several typical methods to evaluate the model performance, so that researchers can test whether the datasets they choose are effective in model training.

Table 2. Relation Between Datasets and Models

Dataset	Model
Common Crawl	GPT-3, LaMDA, LLaMA, Gopher, Chinchilla
Wikipedia	GPT-3, LaMDA, LLaMA, PaLM, Gopher, Chinchilla
News	PaLM
ArXiv	LLaMA
Dialogue forums (e.g., Stack Exchange)	LLaMA, LaMDA, PaLM
Books	GPT-3, LLaMA, PaLM, Gopher, Chinchilla
Code sources (e.g., GitHub)	LLaMA, PaLM, Gopher, Chinchilla, LaMDA

1) *Language Tasks*

Neatness. Wikipedia is used in all six models, showing that it is relatively important in the pre-train process. Researchers consider Wikipedia as a high-quality dataset given the neatness of its content and use it to increase the overall quality of the training set.

Variety. The LLaMA team has also discovered that including different versions of Common Crawl data offers the model various knowledge in the pre-train process and will enhance the final performance of pre-train LLMs[39].

Chain of Thoughts. Including codes is expected to benefit models' chain-of-thought performance[65].

Coherence. Books data is stated to be helpful to long-range context and coherent storytelling[52].

Evaluation. The evaluation of NLP task performance is a popular topic. This paper introduces four famous open-sourced test sets: SQuAD[66], GLUE[67], TruthfulQA[68], and RealToxicPrompts[69]. SQuAD is a reading comprehension dataset that contains 100k+ questions based on Wikipedia articles. GLUE is a collection of existing datasets and is used to evaluate natural language understanding systems. SQuAD and GLUE test the general natural language understanding ability of LLMs. Compared with these two datasets, TruthfulQA and RealToxicPrompts are test sets built specifically for measuring truthfulness and toxicity. TruthfulQA contains 817 questions designed for testing imitative falsehoods which also implies the robustness of a model. RealToxicPrompts contains 100k prompts and toxicity scores for evaluating the toxicity of outputs like racist language.

Besides using the suggested test set to evaluate the performance of pre-train language models, another innovative way is to use the existing superior model to evaluate the current model. The Vicuna team introduces this method as a novel method for language model evaluation[70].

In the meantime, it is worth noting that making conclusions about the effectiveness of the dataset merely by model performances on a test set might not be convincing. It would be hard to determine whether the differences in performance are caused by the data or other factors like model architecture. Previous studies have discovered the influence of pre-train datasets in general by analyzing the model output and the frequency of the knowledge appearances in pre-train datasets[71, 72], but how to figure out the influence of data given the evaluation result is still a problem.

2) *Downstream Tasks*

Pre-train LLMs have the capability to present what are given to them regarding performing downstream tasks in different subjects. This is reflected in several scenarios:

News Summarization. Recently news summarization is a popular desired downstream task of LLMs and can probably be benefited by including news in the pre-train process[73, 74].

Scientific Writing. ArXiv has been included in the Pile dataset with expectations of writing scientific papers[52].

Multitask Understanding (Not Specify Tasks). A

potential explanation of the performance of LLaMA[39] states that ArXiv and book data also allow LLMs to perform well in massive multitask language understanding.

Communication. Introduced by LaMDA[32], high-quality dialogue forums like Stack Exchange are used in the pre-train process of LLMs to let the model have a great performance on chatting as well as maintaining the other abilities of LLMs. Current challenges in communication tasks include cross-domain coherence and open-domain questions. In reality, a dialogue will transit across domains, and thus the model needs to achieve a natural transition while the topic shifts[75]. Open-domain questions refer to the problem that the language model might over-focus on the text and ignore the important information carried from layout or images when answering open questions regarding given documents[76].

Coding. Coding is often an expected downstream ability of LLMs, as many teams hoping to train a model have great abilities to generate and interpret code[77, 78]. A typical code source is GitHub (Google BigQuery).

Mathematical Reasoning. Mathematical reasoning as a downstream task is closely related to the model's language ability about chain-of-thought[49]. Thus evaluating the model on mathematical reasoning can reflect its cognitive ability. A useful evaluation dataset is MATH[79], which has been used in testing LLaMA. This task is difficult for abundant reasons. For example, answering mathematical questions needs a comprehensive understanding of heterogeneous data, which might not be presented in text or numbers[80]; there is also a "bridge between learning and applying" regarding mathematical logic[81]. Typical hard problems in math for machines include geometric problem-solving and proving in general. Both tasks require superior logical reasoning skills. Researchers are actively seeking approaches to improve the model performance like self-enriching the learned library[82]. Though it is not common to include math datasets directly in the pre-train process, it has been figured out that including code source may be beneficial to the model's ability on math since there have been several methods which encourage including code source to solve math problems[83]; there is also evidence that the usage of synthetic data (in proof) will improve the natural language model's ability in reasoning skills, specifically for geometry[84].

Evaluation. A common evaluation open-sourced dataset for downstream tasks is MMLU[85]. This manually created test set contains multiple choice questions from various domains including humanities, social sciences, hard sciences, and other areas that might be interesting to researchers. MMLU is a well-known test set for measuring the multitask performance of LLMs and has been used in famous pre-train LLMs like LLaMA[39] and PaLM[49].

3.1.3 Data Usage

LLMs are token-hungry in pre-train processes and require high-quality corpus[5]. Based on our research, cleanness, diversity, and quantity are three common points when researchers construct datasets that satisfy both the quality and quantity requirements.

1) *Cleanness.* Cleanness refers to whether the data is after filtering or not. Several recent studies have shown that models trained with filtered data can give better results (for example more logistical and harmless) compared with raw data[38, 86]. Thus researchers prefer using clean datasets when training models[87]. Using this metric, some datasets have high-quality data by default because they are extracting data from neat sources like Reddit or Wikipedia. Some datasets are based on raw data, but they did some operations to ensure that most of the bad data is filtered out, like C4[55]. These kinds of datasets are not considered high-quality data because they may contain some bad corpus that has not been filtered.

2) *Diversity.* Diversity refers to the richness of topics in texts. It has been mentioned in several pre-train models like GPT-3 and OPT[30, 88]. A recent study shows that the diversity increases as the number of latent concepts increases, intuitively[59]. Thus web data is more diverse compared with one-source data like Wikipedia. Researchers can also use combined datasets from different sources to yield a better result, but it should also be noticed that simply packing datasets might not increase the overall diversity and may cause unnecessary costs in the training process[89].

3) *Quantity.* Quantity is also an important aspect of training data. Both zero-shot and one-shot performances (evaluated by Natural Questions dataset[90]) of models are highly related to the datasize[39]. The proper quantity of data for pre-train is well studied these years[91]. The two most famous theories that are widely adopted in model training are Scaling Law[4] and Chinchilla Law[50]. The two teams came up with equations with different powers representing the relation between the number of parameters (N) and the

amount of data (D). The Scaling Law proposes that as the number of parameters increases, the data size should be increased with $N^{0.74}$, while the Chinchilla Law proposes that N and D should be scaled equally to get the optimal result[4, 50]. Both the Scaling and Chinchilla laws emphasize that training pre-train models needs an appropriate amount of data and the lack of data will hurt model performance.

The paper compares the size of some datasets from Common Crawl and neat sources like the Gutenberg Project and visualize them in Fig.3. It shows that the size of datasets derived from neat sources is much smaller. This reflects the shortage of high-quality data on the Internet. Thus LLM developers need to balance the quality and quantity of data in the training set when preparing data. Table 3 lists the datasets that are used in some pre-train models as precise examples for researchers. The table discards the Gopher model because Chinchilla includes all the datasets used in Gopher.

3.2 Fine-Tune Datasets

The fine-tune data has a direct influence on the model and is used to improve a specific downstream task. Compared with the structure of the previous section, this subsection deletes the part that talks about the function of datasets and suggest some considerations about the construction of data instead.

3.2.1 *Datasets for Fine-Tune Models*

For fine-tune models designed for dialogue, the fine-tune process often focuses on increasing the quality, ground-truthness, and safety of the model to make it more helpful and align with human values[31, 32]. Researchers have done lots of work on constructing datasets to achieve this goal. This subsection will present fine-tune datasets in three categories: instruction-tuning data, comparison data (human feedback data), and shared conversations. Fig.4 shows fine-tune datasets in different categories, and Table 4 summarizes our surveyed fine-tune datasets.

1) *Instruction Tuning Data*

The outstanding performance of ChatGPT and InstructGPT[31] tells us using instruction tuning will yield a great result. However, purely human-constructed data is hard to gather in reality[92]. To solve this problem, many teams choose to use existing LLMs to help them generate data. Using LLM to con-

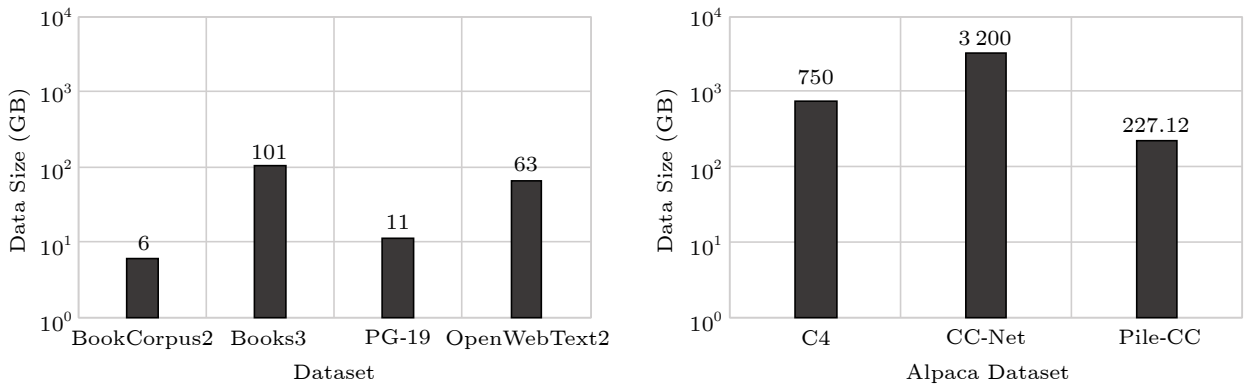


Fig.3. Size of datasets that are constructed by data from (a) high-quality sources and (b) Common Crawl.

Table 3. Pre-Train Data Usage

Model	Data	Brief Explanation
GPT-3	Common Crawl (filtered), Wikipedia, WebText, Books1 and Books2	Include massive web data with some high-quality datasets
LaMDA	C4, Wikipedia, code documents, conversations	Include conversations to increase the chat ability while remain the capability of performing other tasks
LLaMA	English Common Crawl, C4, Wikipedia, Stack Exchange, Gutenberg and Books, ArXiv	Include different versions of Common Crawl for variety and include ArXiv data benefiting multitask language performance
PaLM	Wikipedia, filtered webpages, news, social media conversations, books, GitHub	High-quality corpus without repeating
Chinchilla	MassiveWeb, C4, books, news, GitHub, Wikipedia	Use different filtered datasets with weights

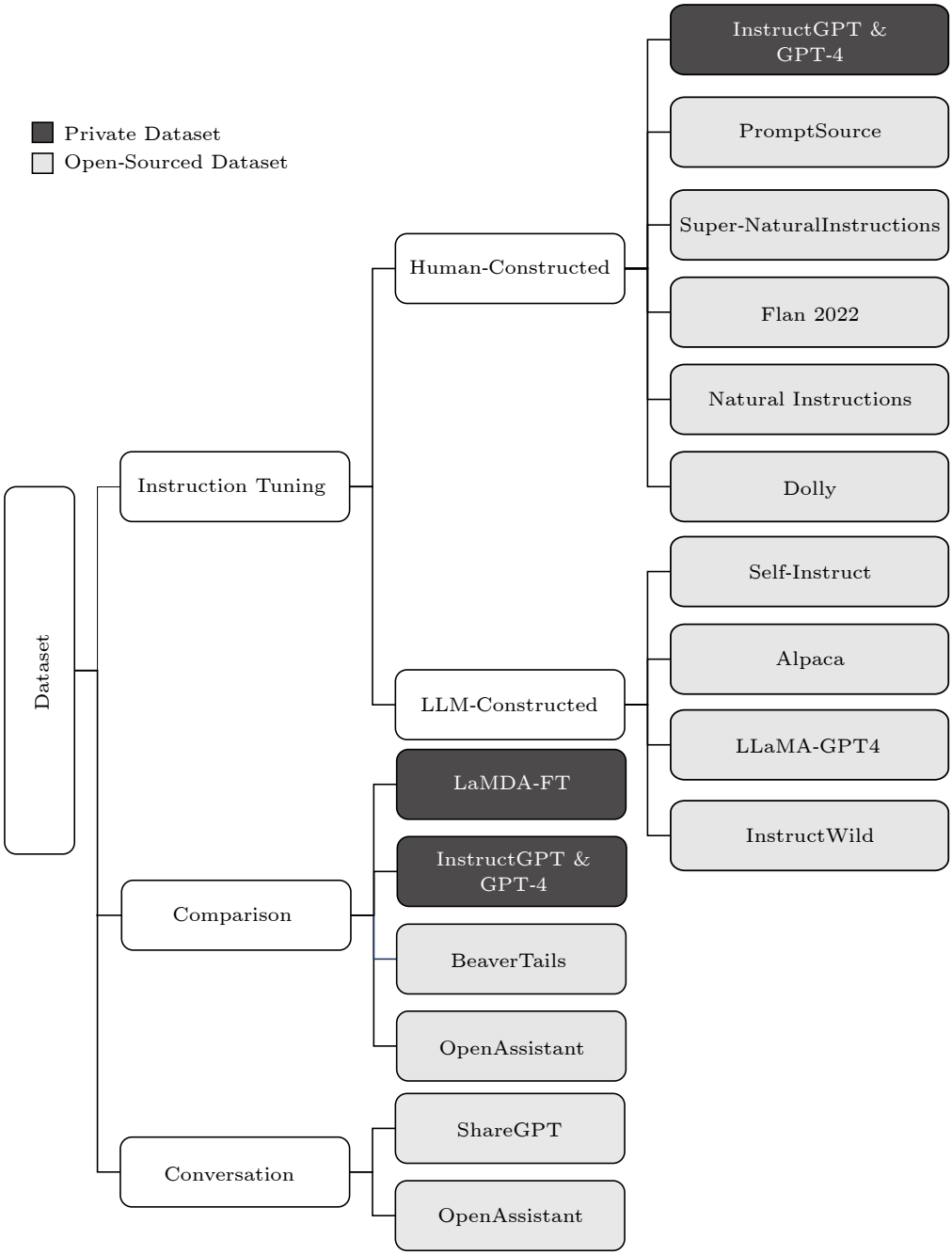


Fig.4. Categories of fine-tune datasets.

struct data has another advantage of ensuring the diversity of the data as human-produced data might contain some patterns due to common preferences among individuals[51]. It is also affordable for most research institutions[2, 51]. This part introduces both human-constructed data and LLM-generated data used in the fine-tune process of several popular LLMs including LaMDA-FT[32], InstructGPT[31], GPTself-inst[51], GPT-4[33], Alpaca, and Vicuna[34]. There are some other well-performed chat models using instruct-tuning like Claude[7], but since the teams do not produce a clear description of the training process, the paper does not include them in this part. Fig.5 illustrates the structure of instruction-tuning, which divides the task into three parts: instruction, input, and output.

[7]https://www.anthropic.com/index/introducing-claude, May 2024.

Table 4. Summary of Fine-Tune Data

Model	Data Size	Data Type	Gathering Method
LaMDA-FT	18.4k dialogues	Labeled dialogue turns	Human labeled
InstructGPT, GPT-4	13k training prompts, reward model: 33k training prompts	Labeled (rating) data, instruction-following	Human labeled
BeaverTails	330k annotated QA pairs	Labeled (rating) data	Human-labeled
OpenAssistant	10k+ annotated conversation trees	Labeled (rating) data, dialogue turns	Human labeled
PromptSource	2k+ prompts	Prompt data	Human-created
Super-NaturalInstructions	1 616 tasks	Instruction-following	Human-created
Flan 2022	1 836 tasks	Instruction-following	Human-created
Natural Instructions	620k tasks	Instruction-following	Human-created
Dolly	15k tasks	Instruction-following	Human-created
LLaMA-GPT4	52k instructions	Instruction-following	ChatGPT synthetic instructions and GPT synthetic answers and comparison data (training reward model)
Alpaca	52k instructions	Instruction-following	Self-instruct based on GPT-3.5 text-davinci-003
Self-Instruct	52k instructions	Instruction-following	Self-instruct based on GPT-3 vanilla
InstructWild	110k tasks	Instruction-following	Manually created instructions and ChatGPT synthetic response
ShareGPT	70k conversations	Dialogue turns	Shared conversation with ChatGPT

The paper first introduces human-constructed datasets.

InstructGPT and GPT-4. The paper discusses InstructGPT[31] and GPT-4 together because the GPT-4 technical report[33] mentions that the fine-tune method for GPT-4 is similar to those for InstructGPT and ChatGPT. Both InstructGPT and GPT-4 use instruction-followed data for supervised fine-tune (SFT). InstructGPT uses 13k human-labeled training prompts.

PromptSource. PromptSource[93] is an open-sourced dataset that contains over 2 000 human-written prompts. This dataset was originally designed for zero-shot learning. The paper includes this dataset because prompt engineering is now an important area in constructing fine-tune datasets for chat models. Researchers can construct instruction-following data with human reflection or LLM synthetic data using this dataset.

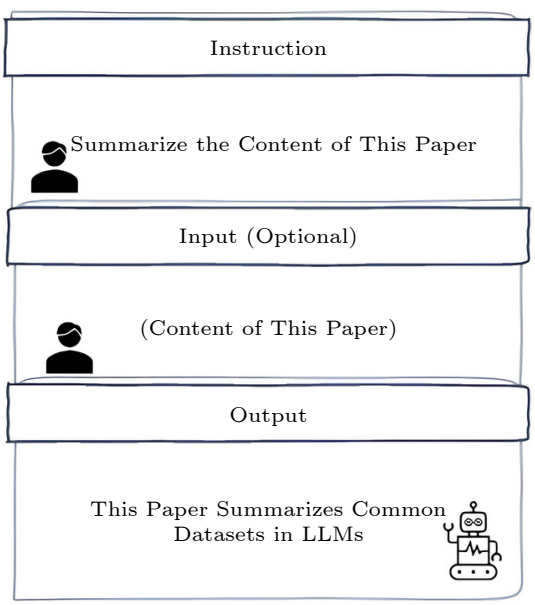


Fig.5. Task components.

Super-NaturalInstruct. Super-NaturalInstructions[94] is a meta-dataset containing 1 616 NLP tasks with instructions. The dataset is multilingual and spans 55 different languages. Similarly to Natural Instructions, it also has both positive and negative examples. It contains instruction, input, and output for each task. The test model Tk-Instruct shows that more observed tasks improve the generalization and simply using a large number of training instances does not help generalization, which emphasizes the importance of diversity for fine-tune datasets.

Flan 2022. Flan 2022[95] is an improved dataset from Flan 2021[96]. This paper only includes Flan 2022 to avoid redundancy. Flan 2022 includes 1 836 multilingual tasks with both zero- and few-shot prompts, as well as tasks designed for chain-of-thought training. Using the Flan 2022 dataset for fine-tune shows that mixed zero- and few-shot prompts can improve performance in both settings.

Natural Instructions. Natural Instruction[97] is a large curated instruction containing various reasoning skills. It consists of 61 subtasks and 620k instances, with both positive and negative responses. The dataset contains prompt, input, and output,

which can be easily transformed into formal instruction-following instances. Experiments show that the model after fine-tuning with natural instructions has generalization to unseen tasks.

Dolly. databricks-dolly-15k[8] is a human-constructed instruction-following dataset. It contains a 15k corpus generated by Databricks employees. The resulting model Dolly and Dolly2 show that the old model fine-tuned on instruction-following data can also have significant improvements. Though Dolly and Dolly2 are not state-of-the-art models, they successfully demonstrate the effectiveness of instruction-tuning.

The paper then introduces LLM-constructed datasets.

GPT Self-Inst. GPT Self-Inst is a dataset with 52k LLM-synthetic instructions generated by GPT-3. GPT Self-Inst uses a self-instruct pipeline to help researchers get data easily using only a few amounts of human-produced data and a model from the GPT-3 series. The pipeline is illustrated in Fig.6. This pipeline requires a few human-produced tasks and then generates new tasks using the LLM that will be fine-tuned. The tasks are divided into classification tasks and non-classification tasks for which classification task instances (input and output) will be generated in the output-first order and non-classification tasks will be generated in an input-first order. The team only uses 175 initial tasks (labeled by humans) as seeds and uses the vanilla GPT-3 model to generate new instructions.

Alpaca. Alpaca is a similar dataset with 52k LLM-synthetic instructions, but this time the data is generated by GPT-3.5. The Alpaca team uses the self-instruct pipeline to generate data, but they make some modifications to make the fine-tune progress more efficient. Instead of vanilla GPT-3, the Alpaca team uses the self-instruct pipeline on GPT-3.5 text-davinci-003 to generate data. They also use a new prompt to make the request more explicit and generate more instructions every time. For the process of generating input and output, the Alpaca team chose to not differentiate between classification tasks and non-classification tasks.

LLaMA-GPT4. The datasets contain English instruction-following data, Chinese instruction-following data, and comparison data. The two instruction-following datasets are based on the 52k instructions generated by the Alpaca team, with the modification that the outputs are built by GPT-4 and Chinese instructions are translated by GPT-4.

InstructWild. The latest version of InstructionWild[9] contains over 110k high-quality user-based instructions. The data is constructed using ChatGPT and has both English and Chinese versions. InstructWild collects instructions from various websites including Twitter, GitHub, Cookup.AI, and Discord. It does not use LLM to generate instructions. The pipeline of using LLM to generate responses is the same as the Alpaca.

2) *Comparison Data (Data with Rating/Feedback)*

Using human feedback is a classic way of fine-tune chat models. It further aligns LLM with human values and improves general LLM benchmarks like safety[32, 98].

LaMDA-FT. The goal for LaMDA fine-tune is to increase the model quality, safety, and groundedness. The three datasets used to train on these aspects are binary labels to dialogues about sensible, specific, and interesting, binary labels to dialogues about safety,

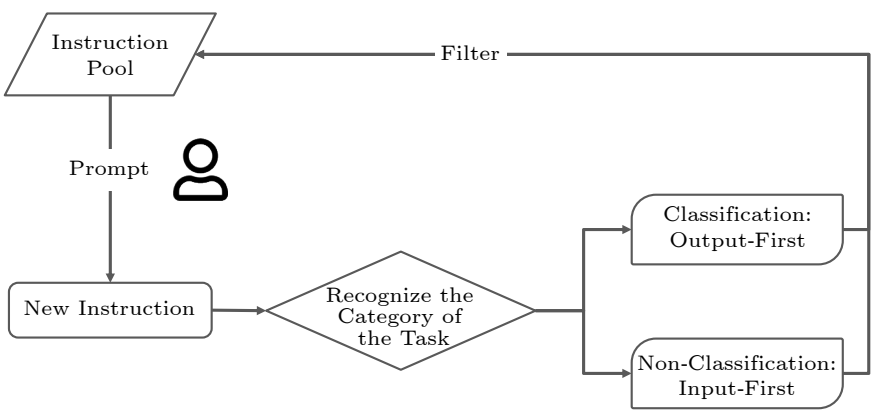


Fig.6. GPT Self-Inst pipeline[51].

[8]https://www.databricks.com/blog/2023/04/12/dolly-first-open-commercially-viable-instruction-tuned-llm, May 2024.
[9]https://github.com/XueFuzhao/InstructionWild, May 2024.

and human evaluations to model responses based on informativeness and groundedness.

InstructGPT and GPT-4. In addition to SFT, InstructGPT and GPT-4 also trained a reward model (RM) using human rating data, which is further used in proximal policy optimization (PPO) to select outputs from a set of candidates. SFT, RM, and PPO together form the three steps of the fine-tune process: supervised training using labeled data, training reward model, and using outputs from the reward model in reinforcement learning.

BeaverTails. BeaverTails[98] is an annotated dataset and the base data type is QA pairs. It has two versions. BeaverTails-30k assigns each QA pair to one crowdworker, whereas Beaver-330k assigns each QA pair to multiple crowdworkers and receives 3.34 annotations for each pair on average. The dataset has been used to train Safe-RLHF, which shows improvement in safety compared with the baseline model, Alpaca7B.

OpenAssistant. OpenAssistant[99] is a human-created and human-annotated corpora with over 10k fully annotated conversation trees.

LLaMA-GPT4. The comparison data is composed of ratings from GPT-4 to its own response. The ratings are from 1 to 10. The team also let GPT-4 rate the responses generated by other models as a comparison.

3) *Conversation*

Recently, online conversations are also used as effective fine-tune data as presented by models like Vicuna[34]. Compared with the first two types of data, using conversation is not a common fine-tune method.

ShareGPT. ShareGPT.com is a website that contains interesting dialogues with ChatGPT shared by customers. The Vicuna team uses fine-tune LLaMA using conversation data from this website and gets good results after evaluating using GPT-4.

OpenAssistant. OpenAssistant is a dataset based with over 161k messages. It contains multiple languages and is dominated by English and Spanish.

3.2.2 Considerations About LLM Synthetic Data

Unlike pre-train datasets that are gathered using similar ways, fine-tune datasets are produced from various methods that bring some inner thoughts. LLM investigators need to carefully examine those different methods when considering what data should be used in models.

The process of collecting human-construct fine-tune data is empirically challenging[100, 101]. As most of the teams cannot gather pure human-labeled data, using LLM is now a widely adopted method to generate instruction-following data[51] and comparison data[45]. This method faces several problems that are worth considering:

1) *Knowledge Distillation.* Researchers can choose various LLMs to generate data, and surely there are some differences between using pre-train LLMs and fine-tune LLMs. Although using fine-tune LLMs like ChatGPT offers great performance[34], it is doubted whether they are distilling knowledge from those models to their models[102, 103]. A potential threat of this problem is that the newly created fine-tune model has no improvement of intrinsic abilities as LLMs[104]. Instead, it relies on and inherits traits from the existing fine-tune models that are trained based on human-produced data[102]. To solve this problem, a current study proposes a new framework with almost no dependency on the teacher model[105].

2) *Comparison Data Quality.* Using LLM to generate comparison data requires the model to find the disadvantages of outputs and rank the outputs fairly. Recent studies show that state-of-the-art LLMs can self-refine and self-verify[106, 107], which reflects that LLMs have a certain level of introspection, but there is still a noticeable deficiency between them with the self-knowledge level of human[108]. The fairness of LLMs is also under-challenged as they might suffer from problems like positional biases[109]. As the ability of LLMs as evaluators is not certain yet, researchers should consider the potential problems of using LLMs as evaluators to produce comparison data.

3) *Prompt Engineering.* Using the LLM synthetic data also requires a careful choice of prompts that instruct LLM to produce instructions (optional), inputs (optional), and outputs. Recent studies show that differences in the narrative of the prompt may result in different model outputs[110, 111]. The main method of generating instructions is to give the LLM some examples and then let it generate a series of instructions, and there are various ways to narrate it. Researchers need to think about how to narrate to get a good result. The Stanford Alpaca team recommends using explicit prompts as it will make the data more diverse.

3.2.3 Data Usage

This subsection gives brief guidance on how to use

the existing datasets in the fine-tune process as well as some points that are worth considering when constructing new fine-tune datasets or using existing datasets.

1) *Fine-Tune Methods*

There are two typical ways of fine-tuning a chat model. Since this paper focuses on data, it only provides a brief description of the pipeline of the training method and suggests some datasets that can be used in these pipelines.

Instruction-Tuning. Introduced by InstructGPT, instruction-tuning is now a popular method of fine-tuning chat models. A formal task in instruction tuning has three parts: instruction, input, and output. The tasks are then sent to pre-train LLMs for supervised learning. Instruction-following data can be created from different methods, including formatting conversation and LLM synthesis[112]. Vicuna uses 70k conversations from SharedGPT.com and gets 90% ChatGPT quality according to GPT-4. Similarly, OpenAssistant and Dolly also get good performance using conversation data. LLM synthesis data is another efficient way for constructing instruction-following data. Well-known chat models like Alpaca and GPT-Self-Instruct all use LLM-constructed instruction data in their fine-tune process.

Supervised Training/Reward Model. Using human-labeled text data to perform supervised training is also a typical way of fine-tuning a chat model. In LaMDA-FT, researchers constructed three datasets with binary labels to improve the model's quality, safety, and groundedness. After this, InstructGPT and some other LLMs show promising results using Reinforcement Learning from Human Feedback (RLHF), which is used to train reward models[112]. Although the importance of RLHF is generally admitted, it is hard for researchers to operate in reality due to the difficulty of gathering human-rated data. This paper introduces two existing open-sourced human-rated datasets, BeaverTails[98] and OpenAssistant[99], to support public researchers training reward models.

2) *Dataset Creation and Usage*

Previous work pointed out that the two directions of fine-tuning chat models are expanding task diversity and offering human reflections[95]. This part discusses strategies for creating and using datasets from these two aspects, as well as the quantity of fine-tune data.

A traditional way of creating fine-tune data is just gathering data from human beings. However, as suggested by GPT Self-Inst, pure human-generated data may suffer from the potential bias from human beings and hurt the overall task diversity[51]. One of the explanations for this is humans are struggling to produce highly complex instructions, but machines can be tuned to generate data with various complexity and thus perform better on a wide range of tasks[113]. The advantage of pure human-generated data is its reality, in other words, accuracy. A recent study points out the accuracy-diversity trade-off and the authors suggested that they improved the performance by using both human-generated data and LLM-generated data[114]. Thus when constructing fine-tune data, researchers might want to include both human data and LLM synthetic data in their training set. Alignment with human value is another essential point of fine-tune datasets, and a promising way to achieve this is to use the RLHF method[112].

Regarding the scaling law of fine-tune, the dataset in this step is commonly agreed to have a much smaller data size[115]. The relation between fine-tune data size and model performance has been studied several times (e.g., studies about data size and model probing performance for BERT[116], and influence of the number of instruction tasks to model scaling up to 1.8k data[117]). However, none of them reflects information about today's state-of-the-art models due to the different model architectures (decoder-only transformers) and the limited amount of fine-tune data (the amount of instruction-following fine-tune data used today is 52k in Alpaca and LLaMA-GPT4). The relation between the data size and the number of parameters for optimal performance in the fine-tune process is unclear now, but researchers can gain an insight by comparing the fine-tune data size used for the Vicuna model and the data size for Alpaca (they are both fine-tuned on LLaMA and give different results).

Vicuna-13B uses more than 10 times of data compared with Alpaca-6B and outperforms it on many benchmarks (Fig.7). This implies that fine-tune chatting models may perform better when having a larger model size and data size.

4 Visual Case

This section provides a brief pipeline with visual cases of training LLMs from an autoregressive model to chatbot, focusing on the aspect of data usage.

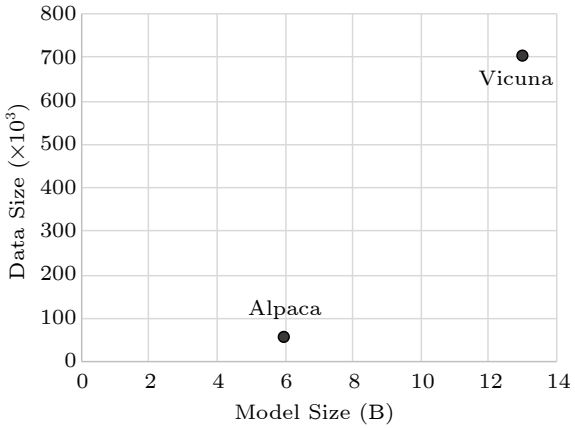


Fig.7. Data size for Alpaca and Vicuna. The y-axis represents the amount of instruction-following data. Vicuna uses 70k conversations which are approximately 700k pieces of instruction-following data.

4.1 Pre-Train Step

The goal of the pre-train process is to get a decoder model that has the ability to perform general NLP tasks like text generation. Before constructing the dataset, there are several points worth to consider.

1) *Language*. Recently, most of the pre-train datasets are English (as shown in Fig.8). Thus it is important to contain some multilingual datasets if you want your model to perform tasks not in English.

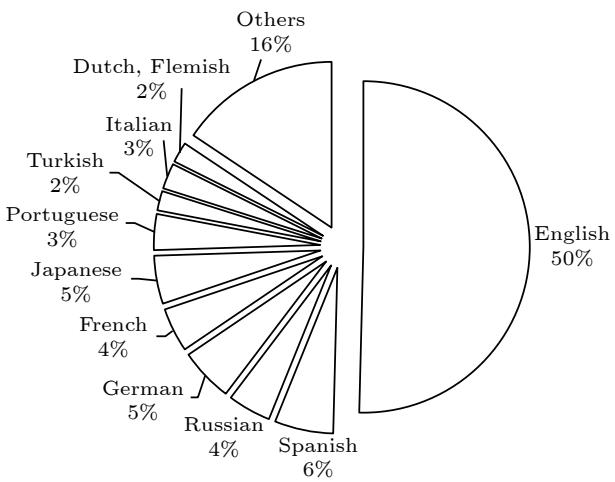


Fig.8. Proportions of websites using different languages[10] (May 2024).

2) *Domain*. As described in Subsection 3.1.2, pre-train LLMs have a general expectation to perform well on domain-specific tasks if they have that knowledge in their pre-train dataset. Thus if you want your pre-train LLMs to perform well on a specific domain, you may want to include the data in the pre-train dataset.

After that, the training set can be constructed through the following two steps.

1) *Data Preparation*. When gathering datasets, remember to ensure the cleanness and diversity of data. To achieve this goal, you need to combine large datasets containing web texts with high-quality datasets together. For example, you can combine C4 and English Wikipedia together as training corpora. The large C4 dataset ensures the diversity of pre-train data as a web dataset, and English Wikipedia can increase the overall cleanness of datasets[39]. Based on your purposes, you can also include other multilingual and domain-specific datasets like the medical subset in the Pile. To further refine the dataset, you can include books and coding data, as they have been used to enhance chain-of-thought and coherence in previous studies[52, 65].

2) *Data Augmentation*. A common strategy used in pre-train is to change the weights of different categories of data by repeating important and less high-quality data[30, 32]. For example, you can augment the Wikipedia subset since Wikipedia is a high-quality dataset but is small in size. This improves the overall quality of the pre-train dataset. Researchers have been concerned about the overfitting problem of pre-train models on high-quality data, but this is then considered as a necessary exchange for better output[30].

4.2 Fine-Tune Step

For a fine-tune model, you can use both instruction-tuning and RLHF techniques so that the model is capable of answering diverse questions with aligned human values. The training set can be constructed through the following two steps.

1) *Instruction Data*. Let us mix LLM-generated and manually created data to form our instruction-following dataset for better performance[114]. The LLM synthetic data enriches the diversity of instruction datasets, which is then useful for fine-tune models to perform various tasks. The existing LLM synthetic datasets like LLaMA-GPT4 can be used. Besides, LLM synthetic data can be created using existing well-performed LLMs like GPT-3.5.

2) *Comparison Data*. The RLHF technique needs some data with human ratings to train the reward

[10] https://w3techs.com/technologies/overview/content_language, May 2024.

model. This allows the model's outputs to align with human values[112]. The comparison data can also be LLM generated or manually created. Different construction methods of comparison data can result in similar performance.

5 Problems and Trends

Previous sections of this paper summarize the datasets used in LLM training and provide a brief analysis regarding the datasets' content and their relation with the models. This section will first offer some comments regarding the current problems and then point out the possible trend of developments of LLM datasets.

Labor cost is a problem faced by all public LLM researchers and developers. They are actively shifting their eyes from costly human-instruction to the cheaper "LLM-instruction".

5.1 Under-Performance on Non-English Data

One problem exists in current LLM datasets is the lack of multi-language corpora. As shown in Table 1, most of the pre-train datasets are composed primarily in English because it is the dominant language for many websites. Fig.8 also shows that the English data has a significant proportion in NLP corpora[11].

As a result, some models trained on those English datasets can perform well on English tasks but not on jobs in other languages. For example, ChatGPT as one of the most powerful LLM performs significantly worse than mT5[118] in multilingual tasks[119], and it has worse performance on non-English text interpretation comparatively[1]. This doubts the reasonability of using or fine-tuning a model trained by English corpora to perform tasks in other languages. Though there is an increasing number of large datasets containing non-English corpus these days (e.g., [53, 120]) and some researchers are working on producing open-sourced multilingual models (e.g., [36]), the big gap between the number of corpora in English and in other languages can hardly be solved in a short time. As a result, English LLMs will dominate the community for a long time, and fine-tuning pre-train LLMs in other languages will not give researchers ideal performance.

5.2 Lack of Open-Sourced Professional Content

Another obstacle is that open-sourced pre-train datasets cannot support LLM developers' increasing demands for professional content. From 2020 to 2023, pre-train models started to include professional content that benefits specific downstream tasks, such as math and code. Table 5 shows that from May 2020 to March 2023, there were more domain-specific data included in the pre-train datasets for LLMs. This is probably because including domain knowledge can enhance the language model's performance in the expected downstream tasks[121]. The open-sourced professional data with high quality is limited, which can hardly support this increasing demand. It is worth noting that facing this challenge, OpenAI chooses to receive help from third parties to enlarge their pre-train dataset[33].

Table 5. Timeline of Data for Pre-Train LLMs

Model	Time	Data
GPT-3[30]	May 2020	General
Gopher[40]	Dec. 2021	General, code
LaMDA-PT[32]	Jan. 2022	General, dialogue, code
PaLM[49]	Apr. 2022	General, dialogue, code
OPT[88]	May 2022	General, dialogue, math
LLaMA[39]	Feb. 2023	General, code, ArXiv, dialogue
PanGU-$\sum$[122]	Mar. 2023	General, domain data, code

Note: General data includes data from webpages (including Wikipedia and news) and books.

5.3 Difficulty for Public Chat Models to Catch up with Industry Models Regarding Fine-Tune Methods

The confidentiality of industries' fine-tune methods is always a problem faced by public researchers. The data construction methods of most chat models mimic the methods of industry models. From LaMDA-FT to ChatGPT, the data collection method for fine-tune models keeps updating in industries. In contrast, most public researchers are imitating instruction-following data (e.g., [123–126]) because of its success in InstructGPT[31].

This imitation is reasonable because most public researchers or academic institutions are incapable of supporting experiments of collecting data creatively in training LLMs. The development of LLMs relies on

[11] https://w3techs.com/technologies/overview/content_language, May 2024.

huge computing resources[127], which can be hardly achieved by individual or non-profit researchers. However, imitation may let private (industry) models continuously leave behind public models. It has been shown that data collection is an important step to improve AI chatbots' performance given this is the only change in the fine-tune method from InstructGPT[31] to ChatGPT, and the top up-to-date chat models tend to not reveal their latest method of constructing training set according to our statistics as shown in Table 6.

Table 6. Timeline of Construction Methods for Fine-Tune Data

Model	Time	Method
LaMDA-FT[32]	Jan. 2022	Labeled dialogue-turns
InstructGPT[31]	Mar. 2022	Instruction-following data
ChatGPT	Nov. 2022	Instruction-following data mixed with new dialogue dataset (no clear description)
Bard[12]	Feb. 2023	No-description
GPT-4[33]	Mar. 2023	No-description
Claude	Mar. 2023	No-description
Claude2[13]	Jul. 2023	No-description

Thus public researchers can hardly get comparable models merely by imitation like before (e.g., GPT Self-Inst[51] gets comparable performance as InstructGPT[31]). Several evaluations suggest that public chat models fine-tuned using instruction data cannot compete with models using novel data collection methods[128, 129]. Thus a potential problem in the research community is that the development of public models is restricted by the data collection method revealed by private models, and cannot catch up with industry models in performance.

Besides, many open-sourced fine-tune models use knowledge distillation to get good performance at a low cost (e.g., [34, 45]) through the help of "teacher models" (models used to generate fine-tune data). This method is challenged by the idea that the resulting model cannot exceed its "teacher" in performance[102]. Since those teacher models are normally private models produced by industry, these public models fine-tune through knowledge distillation cannot get competitive performance as industry models.

5.4 Increasing Ethical Concerns Raised in LLM Training

Firstly, problems related to the privacy, security, and fairness of existing corpora are increasingly awared by the public. A large proportion of data is grabbed from the Internet, which might offend personal data privacy. There is also a lot non-opened data that is doubted that whether it is gathered in proper ways. Regarding the security and fairness of data, corpora may contain bias, harmful, or poisonous language even after filtering by the original unfairness or possible intentional data poisoning attack. These are also big problems in model usage. For example, conversations between someone and the model may be included in the training set and will appear in another person's conversation when he or she uses the model, and the model may produce unsafe output when the user intentionally provides poisonous inputs. Researchers are now actively finding method to solve the problems related to fairness and security of datasets. One effective method is red teaming. Basically, red teaming the model refers to the process of mimicking the poisonous conversation in reality to discover and reduce the possible harmful outputs[130]. This method has been used in the production of BeaverTail[98].

Besides, LLM has the property that the model will perform better as if there is more investment[127]. Nowadays LLM pre-train needs a great amount of computing/hardware resources. This prohibits individual or public researchers from training the state-of-the-art models independently. Another concern related to this problem is the negative environmental influence due to high energy consumption[131].

5.5 Increasing Research Interests in LLM-Generated Data

Aside from the problems, there is a trend in LLM dataset development that public researchers are increasingly focusing on LLM-generated data, instead of human-instruction data. The key datasets used in the fine-tune process of GPT and other models having great performance are produced by human labelers. It needs huge investment which is not achievable for public researchers and most academic institutions. Given this background, skipping the heavy labor in

[12] https://bard.google.com, May 2023.
[13] https://www.anthropic.com/index/claude-2, May 2024.

the fine-tune process is now an attractive topic. Many researchers tend to use a more affordable way to construct data, which adopts help from LLM[132, 133]. To visualize the trend of popularity, this part counts and visualizes (Fig.9) the number of papers in the ArXiv hub discussing the usage of "self-instruct" in language models, which is one of the famous strategies to reduce the cost of labor[51].

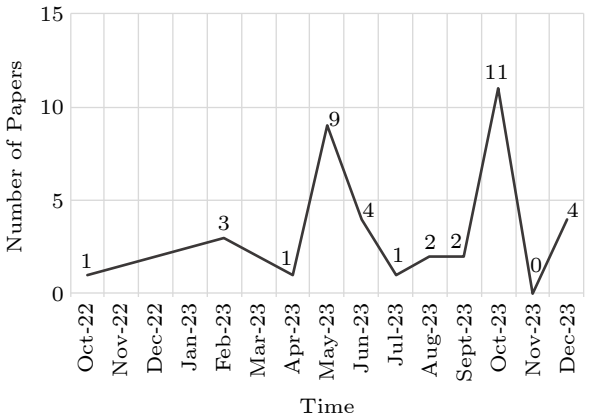


Fig.9. Number of papers searched by "self-instruct" and "language model" in the ArXiv repository.

6 Conclusions

This paper summarized and discussed pre-train and fine-tune datasets used in nowadays popular decoder-only pre-train models and fine-tune chat models, along with brief introductions of the background knowledge including data operation and usage. The paper then provided a visual case for researchers to better understand and figure out problems and possible trends about LLMs.

The limitation of this paper is that it only focuses on datasets and does not discuss the performances of the models. In the future, we would like to examine how differences in datasets influence model performances. Some topics we are interested in include tracing the influences of different datasets in the pre-train dataset by evaluating the model performance, and how different data construction methods in the fine-tuning process influence the model's chatting performance.

Conflict of Interest The authors declare that they have no conflict of interest.

References

[1] Bang Y, Cahyawijaya S, Lee N *et al.* A multitask, multilingual, multimodal evaluation of ChatGPT on reasoning, hallucination, and interactivity. In *Proc. the 13th International Joint Conference on Natural Language and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*, Nov. 2023, pp.675–718. DOI: 10.18653/v1/2023.ijcnlp-main.45.

[2] Zhao W X, Zhou K, Li J Y *et al.* A survey of large language models. arXiv: 2303.18223, 2023. https://arxiv.org/abs/2303.18223, May 2024.

[3] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez A N, Kaiser Ł, Polosukhin I. Attention is all you need. In *Proc. the 31st International Conference on Neural Information Processing Systems*, Dec. 2017, pp.6000–6010.

[4] Kaplan J, McCandlish S, Henighan T, Brown T B, Chess B, Child R, Gray S, Radford A, Wu J, Amodei D. Scaling laws for neural language models. arXiv: 2001.08361, 2020. https://arxiv.org/abs/2001.08361, May 2024.

[5] Xue F Z, Fu Y, Zhou W C S, Zheng Z W, You Y. To repeat or not to repeat: Insights from scaling LLM under token-crisis. arXiv: 2305.13230, 2023. https://arxiv.org/abs/2305.13230, May 2024.

[6] Bai Y T, Jones A, Ndousse K *et al.* Training a helpful and harmless assistant with reinforcement learning from human feedback. arXiv: 2204.05862, 2022. https://arxiv.org/abs/2204.05862, May 2024.

[7] Naveed H, Khan A U, Qiu S, Saqib M, Anwar S, Usman M, Akhtar N, Barnes N, Mian A. A comprehensive overview of large language models. arXiv: 2307.06435, 2023. https://arxiv.org/abs/2307.06435, May 2024.

[8] Hosseini M, Gao C A, Liebovitz D, Carvalho A, Ahmad F S, Luo Y, MacDonald N, Holmes K, Kho A. An exploratory survey about using ChatGPT in education, healthcare, and research. PLOS ONE, 18(10): e0292216. https://doi.org/10.1371/journal.pone.0292216.

[9] Ling C, Zhao X J, Lu J Y *et al.* Domain specialization as the key to make large language models disruptive: A comprehensive survey. arXiv: 2305.18703, 2023. https://arxiv.org/abs/2305.18703, May 2024.

[10] Wu L K, Zheng Z, Qiu Z P, Wang H, Gu H C, Shen T J, Qin C, Zhu C, Zhu H S, Liu Q, Xiong H, Chen E H. A survey on large language models for recommendation. arXiv: 2305.19860, 2023. https://arxiv.org/abs/2305.19860, May 2024.

[11] Wang J J, Huang Y C, Chen C Y, Liu Z, Wang S, Wang Q. Software testing with large language models: Survey, landscape, and vision. arXiv: 2307.07221, 2024. https://arxiv.org/abs/2307.07221, May 2024.

[12] Kasneci E, Sessler K, Küchemann S *et al.* ChatGPT for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences*, 2023, 103: 102274. DOI: 10.1016/j.lindif.2023.102274.

[13] Wang B Y, Xie Q Q, Pei J H, Chen Z H, Tiwari P, Li Z, Fu J. Pre-trained language models in biomedical domain: A systematic survey. *ACM Computing Surveys*, 2024,

56(3): 55. DOI: 10.1145/3611651.

[14] Chang Y P, Wang X, Wang J D *et al.* A survey on evaluation of large language models. *ACM Trans. Intelligent Systems and Technology*, 2024, 15(3): 39. DOI: 10.1145/3641289.

[15] Mohamadi S, Mujtaba G, Le N, Doretto G, Adjeroh D A. ChatGPT in the age of generative AI and large language models: A concise survey. arXiv: 2307.04251, 2023. https://arxiv.org/abs/2307.04251, May 2024.

[16] Liu Y H, Han T L, Ma S Y *et al.* Summary of ChatGPT/GPT-4 research and perspective towards the future of large language models. arXiv: 2304.01852, 2023. https://arxiv.org/abs/2304.01852v1, May 2024.

[17] Zhang C N, Zhang C S, Li C H *et al.* One small step for generative AI, one giant leap for AGI: A complete survey on ChatGPT in AIGC era. arXiv: 2304.06488, 2023. https://arxiv.org/abs/2304.06488, May 2024.

[18] Chen K P, Shao A Q, Burapacheep J, Li YX. How GPT-3 responds to different publics on climate change and black lives matter: A critical appraisal of equity in conversational AI. arXiv: 2209.13627, 2023. https://arxiv.org/abs/2209.13627, May 2024.

[19] Zong M Y, Krishnamachari B. A survey on GPT-3. arXiv: 2212.00857, 2022. https://arxiv.org/abs/2212.00857, May 2024.

[20] Wang H, Hee M S, Awal M R, Choo K T W, Lee R K W. Evaluating GPT-3 generated explanations for hateful content moderation. In *Proc. the 32nd International Joint Conference on Artificial Intelligence*, Aug. 2023, Article No. 694. DOI: 10.24963/ijcai.2023/694.

[21] Fernandes P, Madaan A, Liu E *et al.* Bridging the gap: A survey on integrating (Human) feedback for natural language generation. *Trans. Association for Computational Linguistics*, 2023, 11: 1643–1668. DOI: 10.1162/tacl_a_00626.

[22] De Angelis L, Baglivo F, Arzilli G, Privitera G P, Ferragina P, Tozzi A E, Rizzo C. ChatGPT and the rise of large language models: The new AI-driven infodemic threat in public health. *Frontiers in Public Health*, 2023, 11: 1166120. DOI: 10.3389/fpubh.2023.1166120.

[23] Dillion D, Tandon N, Gu Y L, Gray K. Can AI language models replace human participants? *Trends in Cognitive Sciences*, 2023, 27(7): 597–600. DOI: 10.1016/j.tics.2023.04.008.

[24] Egli A. ChatGPT, GPT-4, and other large language models: The next revolution for clinical microbiology? *Clinical Infectious Diseases*, 2023, 77(9): 1322–1328. DOI: 10.1093/cid/ciad407.

[25] Weidinger L, Mellor J, Rauh M *et al.* Ethical and social risks of harm from language models. arXiv: 2112.04359, 2021. https://arxiv.org/abs/2112.04359, May 2024.

[26] Bender E M, Gebru T, McMillan-Major A, Shmitchell S. On the dangers of stochastic parrots: Can language models be too big? In *Proc. the 2021 ACM Conference on Fairness, Accountability, and Transparency*, Mar. 2021, pp.610–623. DOI: 10.1145/3442188.3445922.

[27] Radford A, Wu J, Child R, Luan D, Amodei D, Sutskever I. Language models are unsupervised multitask learners. 2019. https://openai.com/index/better-language-models/, May 2024.

[28] Devlin J, Chang M W, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Jun. 2019, pp.4171–4186. DOI: 10.18653/v1/N19-1423.

[29] Du Z X, Qian Y J, Liu X, Ding M, Qiu J Z, Yang Z L, Tang J. GLM: General language model pretraining with autoregressive blank infilling. In *Proc. the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, May 2022, pp.320–335. DOI: 10.18653/v1/2022.acl-long.26.

[30] Brown T B, Mann B, Ryder N *et al.* Language models are few-shot learners. In *Proc. the 34th International Conference on Neural Information Processing Systems*, Dec. 2020, Article No. 159.

[31] Ouyang L, Wu J, Jiang X *et al.* Training language models to follow instructions with human feedback. In *Proc. the 36th International Conference on Neural Information Processing Systems*, Nov. 2022, Article No. 2011.

[32] Thoppilan R, De Freitas D, Hall J *et al.* LaMDA: Language models for dialog applications. arXiv: 2201.08239, 2022. https://arxiv.org/abs/2201.08239, May 2024.

[33] OpenAI. GPT-4 technical report. arXiv: 2303.08774, 2023. https://arxiv.org/abs/2303.08774, May 2024.

[34] The Vicuna Team. Vicuna: An open-source chatbot impressing GPT-4 with 90%* ChatGPT quality, 2023. https://lmsys.org/blog/2023-03-30-vicuna/, June 2024.

[35] Xu D K, Yen I E H, Zhao J X, Xiao Z B. Rethinking network pruning – Under the pre-train and fine-tune paradigm. In *Proc. the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Jun. 2021, pp.2376–2382. DOI: 10.18653/v1/2021.naacl-main.188.

[36] BigScience Workshop. BLOOM: A 176B-parameter open-access multilingual language model. arXiv: 2211.05100, 2023. https://arxiv.org/abs/2211.05100, May 2024.

[37] Nobata C, Tetreault J, Thomas A, Mehdad Y, Chang Y. Abusive language detection in online user content. In *Proc. the 25th International Conference on World Wide Web*, Apr. 2016, pp.145–153. DOI: 10.1145/2872427.2883062.

[38] Du N, Huang Y P, Dai A M *et al.* GLaM: Efficient scaling of language models with mixture-of-experts. In *Proc. the 39th International Conference on Machine Learning*, Jul. 2022, pp.5547–5569.

[39] Touvron H, Lavril T, Izacard G *et al.* LLaMA: Open and efficient foundation language models. arXiv: 2302.13971, 2023. https://arxiv.org/abs/2302.13971, May 2024.

[40] Rae J W, Borgeaud S, Cai T *et al.* Scaling language

models: Methods, analysis & insights from training gopher. arXiv: 2112.11446, 2022. https://arxiv.org/abs/2112.11446, May 2024.

[41] Lee K, Ippolito D, Nystrom A, Zhang C Y, Eck D, Callison-Burch C, Carlini N. Deduplicating training data makes language models better. In *Proc. the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, May 2022, pp.8424–8445. DOI: 10.18653/v1/2022.acl-long.577.

[42] Frank M C. Bridging the data gap between children and large language models. *Trends in Cognitive Sciences*, 2023, 27(11): 990–992. DOI: 10.1016/j.tics.2023.08.007.

[43] Shin S, Lee S W, Ahn H, Kim S, Kim H, Kim B, Cho K, Lee G, Park W, Ha J W, Sung N. On the effect of pretraining corpora on in-context learning by a large-scale language model. In *Proc. the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Jul. 2022, pp.5168–5186. DOI: 10.18653/v1/2022.naacl-main.380.

[44] Adiwardana D, Luong M T, So D R, Hall J, Fiedel N, Thoppilan R, Yang Z, Kulshreshtha A, Nemade G, Lu Y F, Le Q V. Towards a human-like open-domain chatbot. arXiv:2001.09977,2020.https://arxiv.org/abs/2001.09977, May 2024.

[45] Peng B L, Li C Y, He P C, Galley M, Gao J F. Instruction tuning with GPT-4. arXiv: 2304.03277, 2023. https://arxiv.org/abs/2304.03277, May 2024.

[46] McKenna N, Li T Y, Cheng L, Hosseini M, Johnson M, Steedman M. Sources of hallucination by large language models on inference tasks. In *Proc. the 2023 Findings of the Association for Computational Linguistics*, Dec. 2023, pp.2758–2774. DOI: 10.18653/v1/2023.findings-emnlp.182.

[47] Deshpande A, Murahari V, Rajpurohit T, Kalyan A, Narasimhan K. Toxicity in chatgpt: Analyzing persona-assigned language models. In *Proc. the 2023 Findings of the Association for Computational Linguistics*, Dec. 2023, pp.1236–1270. DOI: 10.18653/v1/2023.findings-emnlp.88.

[48] Amatriain X, Sankar A, Bing J, Bodigutla P K, Hazen T J, Kazi M. Transformer models: An introduction and catalog. arXiv: 2302.07730, 2023. https://arxiv.org/abs/2302.07730, May 2024.

[49] Chowdhery A, Narang S, Devlin J *et al*. PaLM: Scaling language modeling with pathways. *The Journal of Machine Learning Research*, 2023, 24(1): 240. DOI: 10.5555/3648699.3648939.

[50] Hoffmann J, Borgeaud S, Mensch A *et al*. Training compute-optimal large language models. In *Proc. the 36th International Conference on Neural Information Processing Systems*, Nov. 2024, Article No. 2176. DOI: 10.5555/3600270.3602446.

[51] Wang Y Z, Kordi Y, Mishra S, Liu A, Smith N A, Khashabi D, Hajishirzi H. Self-Instruct: Aligning language models with self-generated instructions. In *Proc. the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Jul. 2023, pp.13484–13508. DOI: 10.18653/v1/2023.acl-long.754.

[52] Gao L, Biderman S, Black S, Golding L, Hoppe T, Foster C, Phang J, He H, Thite A, Nabeshima N, Presser S, Leahy C. The pile: An 800GB dataset of diverse text for language modeling. arXiv: 2101.00027, 2020. https://arxiv.org/abs/2101.00027, May 2024.

[53] Laurençon H, Saulnier L, Wang T *et al*. The BigScience ROOTS corpus: A 1.6TB composite multilingual dataset. In *Proc. the 36th International Conference on Neural Information Processing Systems*, Nov. 2022, Article No. 2306. DOI: 10.5555/3600270.3602576.

[54] Huang G. Network of data: Digital infrastructure. *Communication of the CCF*, 2021(12): 58–60. (in Chinese)

[55] Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y Q, Li W, Liu P J. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 2020, 21(1): 140.

[56] Rae J W, Potapenko A, Jayakumar S M, Lillicrap T P. Compressive transformers for long-range sequence modelling. arXiv: 1911.05507, 2019. https://arxiv.org/abs/1911.05507, May 2024.

[57] Wenzek G, Lachaux M A, Conneau A, Chaudhary V, Guzmán F, Joulin A, Grave E. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proc. the 12th Language Resources and Evaluation Conference*, May 2020, pp.4003–4012.

[58] Penedo G, Malartic Q, Hesslow D, Cojocaru R, Alobeidli H, Cappelli A, Pannier B, Almazrouei E, Launay J. The refinedweb dataset for falcon LLM: Outperforming curated corpora with web data only. In *Proc. the 37th International Conference on Neural Information Processing Systems*, Dec. 2023, Article No. 3464. DOI: 10.5555/3666122.3669586.

[59] Lee A, Miranda B, Sundar S, Koyejo S. Beyond scale: The diversity coefficient as a data quality metric demonstrates LLMs are pre-trained on formally diverse data. arXiv:2306.13840,2023.https://arxiv.org/abs/2306.13840, May 2024.

[60] Lee K, Chang M W, Toutanova K. Latent retrieval for weakly supervised open domain question answering. In *Proc. the 57th Annual Meeting of the Association for Computational Linguistics*, Jul. 2019, pp.6086–6096. DOI: 10.18653/v1/P19–1612.

[61] Yuan S, Zhao H Y, Du Z X, Ding M, Liu X, Cen Y K, Zou X, Yang Z L, Tang J. WuDaoCorpora: A super large-scale Chinese corpora for pre-training language models. *AI Open*, 2021, 2: 65–68. DOI: 10.1016/j.aiopen.2021.06.001.

[62] El-Khair I A. 1.5 billion words arabic corpus. arXiv: 1611.04033, 2016. https://arxiv.org/abs/1611.04033, May 2024.

[63] Kakwani D, Kunchukuttan A, Golla S, Gokul N C, Bhattacharyya A, Khapra M M, Kumar P. *IndicNLPSuite*: Monolingual corpora, evaluation benchmarks and pre-trained multilingual language models for Indian lan-

guages. In *Proc. the 2020 Findings of the Association for Computational Linguistics*, Nov. 2020, pp.4948–4961. DOI: 10.18653/v1/2020.findings-emnlp.445.

[64] Armengol-Estapé J, Carrino C P, Rodriguez-Penagos C, De Gibert Bonet O, Armentano-Oller C, Gonzalez-Agirre A, Melero M, Villegas M. Are multilingual models the best choice for moderately under-resourced languages? A comprehensive assessment for Catalan. In *Proc. the 2021 Findings of the Association for Computational Linguistics*, Aug. 2021, pp.4933–4946. DOI: 10.18653/v1/2021.findings-acl.437.

[65] Wei J, Wang X Z, Schuurmans D, Bosma M, Ichter B, Xia F, Chi E H, Le Q V, Zhou D. Chain-of-thought prompting elicits reasoning in large language models. In *Proc. the 36th International Conference on Neural Information Processing Systems*, Nov. 2022, Article No. 1800.

[66] Rajpurkar P, Zhang J, Lopyrev K, Liang P. SQuAD: 100, 000+ questions for machine comprehension of text. In *Proc. the 2016 Conference on Empirical Methods in Natural Language Processing*, Nov. 2016, pp.2383–2392. DOI: 10.18653/v1/D16–1264.

[67] Wang A, Singh A, Michael J, Hill F, Levy O, Bowman S. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proc. the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, Nov. 2018, pp.353–355. DOI: 10.18653/v1/W18–5446.

[68] Lin S, Hilton J, Evans O. TruthfulQA: Measuring how models mimic human falsehoods. arXiv: 2109.07958, 2022. https://arxiv.org/abs/2109.07958, May 2024.

[69] Gehman S, Gururangan S, Sap M, Choi Y, Smith N A. RealToxicityPrompts: Evaluating neural toxic degeneration in language models. arXiv: 2009.11462, 2020. https://arxiv.org/abs/2009.11462, May 2024.

[70] Zheng L M, Chiang W L, Sheng Y, Zhuang S Y, Wu Z H, Zhuang Y H, Lin Z, Li Z H, Li D C, Xing E P, Zhang H, Gonzalez J E, Stoica I. Judging LLM-as-a-judge with MT-bench and Chatbot Arena. arXiv: 2306.05685, 2023. https://arxiv.org/abs/2306.05685, May 2024.

[71] Kandpal N, Deng H K, Roberts A, Wallace E, Raffel C. Large language models struggle to learn long-tail knowledge. arXiv: 2211.08411, 2022. https://arxiv.org/abs/2211.08411, May 2024.

[72] Razeghi Y, Logan IV R L, Gardner M, Singh S. Impact of pretraining term frequencies on few-shot reasoning. arXiv: 2202.07206, 2022. https://arxiv.org/abs/2202.07206, May 2024.

[73] Xiao L, Chen X L. Enhancing LLM with evolutionary fine tuning for news summary generation. arXiv: 2307.02839, 2023. https://arxiv.org/abs/2307.02839, May 2024.

[74] Zhang T Y, Ladhak F, Durmus E, Liang P, McKeown K, Hashimoto T B. Benchmarking large language models for news summarization. arXiv: 2301.13848, 2023. https://arxiv.org/abs/2301.13848, May 2024.

[75] Zhu Q, Huang K L, Zhang Z, Zhu X Y, Huang M L. CrossWOZ: A large-scale Chinese cross-domain task-oriented dialogue dataset. *Trans. Association for Computational Linguistics*, 2020, 8: 281–295. DOI: 10.1162/tacl_a_00314.

[76] Qi L, Lv S W, Li H Y, Liu J, Zhang Y, She Q Q, Wu H, Wang H F, Liu T. DuReader$_{vis}$: A Chinese dataset for open-domain document visual question answering. In *Proc. the 2022 Findings of the Association for Computational Linguistics*, May 2022, pp.1338–1351. DOI: 10.18653/v1/2022.findings-acl.105.

[77] Zhang J Y, Panthaplackel S, Nie P Y, Li J J, Gligoric M. CoditT5: Pretraining for source code and natural language editing. In *Proc. the 37th IEEE/ACM International Conference on Automated Software Engineering*, Oct. 2023, Article No. 22. DOI: 10.1145/3551349.3556955.

[78] Le H, Wang Y, Gotmare A D, Savarese S, Hoi S C H. CodeRL: Mastering code generation through pretrained models and deep reinforcement learning. In *Proc. the 36th International Conference on Neural Information Processing Systems*, Nov. 2022, Article No. 1549. DOI: 10.5555/3600270.3601819.

[79] Hendrycks D, Burns C, Kadavath S, Arora A, Basart S, Tang E, Song D, Steinhardt J. Measuring mathematical problem solving with the math dataset. arXiv: 2103.03874, 2021. https://arxiv.org/abs/2103.03874, May 2024.

[80] Lu P, Qiu L, Chang K W, Wu Y N, Zhu S C, Rajpurohit T, Clark P, Kalyan A. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. arXiv: 2209.14610, 2023. https://arxiv.org/abs/2209.14610, May 2024.

[81] Liu J Y, Huang Z Y, Zhai C X, Liu Q. Learning by applying: A general framework for mathematical reasoning via enhancing explicit knowledge learning. In *Proc. the 37th AAAI Conference on Artificial Intelligence*, Jun. 2023, pp.4497–4506.

[82] Wang H M, Xin H J, Zheng C Y, Li L, Liu Z Y, Cao Q X, Huang Y Y, Xiong J, Shi H, Xie E Z, Yin J, Li Z G, Liao H, Liang X D. LEGO-Prover: Neural theorem proving with growing libraries. arXiv: 2310.00656, 2023. https://arxiv.org/abs/2310.00656, May 2024.

[83] Wang K, Ren H X, Zhou A J, Lu Z M, Luo S C, Shi W K, Zhang R R, Song L Q, Zhan M J, Li H S. MathCoder: Seamless code integration in LLMs for enhanced mathematical reasoning. arXiv: 2310.03731, 2023. https://arxiv.org/abs/2310.03731, May 2024.

[84] Trinh T H, Wu Y H, Le Q V, He H, Luong T. Solving olympiad geometry without human demonstrations. *Nature*, 2024, 625(7995): 476–482. DOI: 10.1038/s41586-023-06747-5.

[85] Hendrycks D, Burns C, Basart S, Zou A, Mazeika M, Song D, Steinhardt J. Measuring massive multitask language understanding. arXiv: 2009.03300, 2021. https://arxiv.org/abs/2009.03300, May 2024.

[86] Snæbjarnarson V, Símonarson H B, Ragnarsson P O, Ingólfsdóttir S L, Jónsson H P, Þorsteinsson V, Einarsson H. A warm start and a clean crawled corpus – A recipe for good language models. arXiv: 2201.05601,

2022. https://arxiv.org/abs/2201.05601, May 2024.

[87] Ngo H, Raterink C, Araújo J G M, Zhang I, Chen C, Morisot A, Frosst N. Mitigating harm in language models with conditional-likelihood filtration. arXiv: 2108.07790, 2021. https://arxiv.org/abs/2108.07790, May 2024.

[88] Zhang S S, Roller S, Goyal N, Artetxe M, Chen M, Chen S H, Dewan C, Diab M, Li X, Lin X V, Mihaylov T, Ott M, Shleifer S, Shuster K, Simig D, Koura P S, Sridhar A, Wang T L, Zettlemoyer L. OPT: Open pre-trained transformer language models. arXiv: 2205.01068, 2022. https://arxiv.org/abs/2205.01068, May 2024.

[89] Wang X, Zhou W K, Zhang Q, Zhou J, Gao S Y, Wang J Z, Zhang M H, Gao X, Chen Y W, Gui T. Farewell to aimless large-scale pretraining: Influential subset selection for language model. arXiv: 2305.12816, 2023. https://arxiv.org/abs/2305.12816, May 2024.

[90] Kwiatkowski T, Palomaki J, Redfield O *et al*. Natural questions: A benchmark for question answering research. *Trans. Association for Computational Linguistics*, 2019, 7: 453–466. DOI: 10.1162/tacl_a_00276.

[91] Pérez-Mayos L, Ballesteros M, Wanner L. How much pretraining data do language models need to learn syntax? arXiv: 2109.03160, 2021. https://arxiv.org/abs/2109.03160, May 2024.

[92] Ding N, Chen Y L, Xu B K, Qin Y J, Zheng Z, Hu S D, Liu Z Y, Sun M S, Zhou B W. Enhancing chat language models by scaling high-quality instructional conversations. arXiv: 2305.14233, 2023. https://arxiv.org/abs/2305.14233, May 2024.

[93] Bach S H, Sanh V, Yong Z X *et al*. PromptSource: An integrated development environment and repository for natural language prompts. arXiv: 2202.01279, 2022. https://arxiv.org/abs/2202.01279, May 2024.

[94] Wang Y Z, Mishra S, Alipoormolabashi P *et al*. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. arXiv: 2204.07705, 2022. https://arxiv.org/abs/2204.07705, May 2024.

[95] Longpre S, Hou L, Vu T, Webson A, Chung H W, Tay Y, Zhou D, Le Q V, Zoph B, Wei J, Roberts A. The flan collection: Designing data and methods for effective instruction tuning. arXiv: 2301.13688, 2023. https://arxiv.org/abs/2301.13688, May 2024.

[96] Wei J, Bosma M, Zhao V Y, Guu K, Yu A W, Lester B, Du N, Dai A M, Le Q V. Finetuned language models are zero-shot learners. arXiv: 2109.01652, 2022. https://arxiv.org/abs/2109.01652, May 2024.

[97] Mishra S, Khashabi D, Baral C, Hajishirzi H. Cross-task generalization via natural language crowdsourcing instructions. arXiv: 2104.08773, 2022. https://arxiv.org/abs/2104.08773, May 2024.

[98] Ji J M, Liu M, Dai J T, Pan X H, Zhang C, Bian C, Zhang C, Sun R Y, Wang Y Z, Yang Y D. BeaverTails: Towards improved safety alignment of LLM via a human-preference dataset. arXiv: 2307.04657, 2023. https://arxiv.org/abs/2307.04657, May 2024.

[99] Köpf A, Kilcher Y, von Rütte D *et al*. OpenAssistant conversations – Democratizing large language model alignment. arXiv: 2304.07327, 2023. https://arxiv.org/abs/2304.07327, May 2024.

[100] Zhang J, Wu X D, Sheng V S. Learning from crowdsourced labeled data: A survey. *Artificial Intelligence Review*, 2016, 46(4): 543–576. DOI: 10.1007/s10462–016-9491–9.

[101] Ramamurthy R, Ammanabrolu P, Brantley K, Hessel J, Sifa R, Bauckhage C, Hajishirzi H, Choi Y. Is reinforcement learning (not) for natural language processing: Benchmarks, baselines, and building blocks for natural language policy optimization. arXiv: 2210.01241, 2023. https://arxiv.org/abs/2210.01241, May 2024.

[102] Sun Z Q, Shen Y K, Zhou Q H, Zhang H X, Chen Z F, Cox D, Yang Y M, Gan C. Principle-driven self-alignment of language models from scratch with minimal human supervision. In *Proc. the 37th International Conference on Neural Information Processing Systems*, Dec. 2023, Article No. 115.

[103] Dettmers T, Pagnoni A, Holtzman A, Zettlemoyer L. QLoRA: Efficient finetuning of quantized LLMs. arXiv: 2305.14314, 2023. https://arxiv.org/abs/2305.14314, May 2024.

[104] Gudibande A, Wallace E, Snell C, Geng X Y, Liu H, Abbeel P, Levine S, Song D. The false promise of imitating proprietary LLMs. arXiv: 2305.15717, 2023. https://arxiv.org/abs/2305.15717, May 2024.

[105] Kim S, Bae S, Shin J, Kang S, Kwak D, Yoo K M, Seo M. Aligning large language models through synthetic feedback. arXiv: 2305.13735, 2023. https://arxiv.org/abs/2305.13735, May 2024.

[106] Madaan A, Tandon N, Gupta P *et al*. SELF-REFINE: Iterative refinement with self-feedback. In *Proc. the 37th International Conference on Neural Information Processing Systems*, Dec. 2023, Article No. 2019.

[107] Weng Y X, Zhu M J, Xia F, Li B, He S Z, Liu S P, Sun B, Liu K, Zhao J. Large language models are better reasoners with self-verification. arXiv: 2212.09561, 2023. https://arxiv.org/abs/2212.09561, May 2024.

[108] Yin Z Y, Sun Q S, Guo Q P, Wu J W, Qiu X P, Huang X J. Do large language models know what they don't know? arXiv: 2305.18153, 2023. https://arxiv.org/abs/2305.18153, May 2024.

[109] Wang P Y, Li L, Chen L, Cai Z F, Zhu D W, Lin B H, Cao Y B, Liu Q, Liu T Y, Sui Z F. Large language models are not fair evaluators. arXiv: 2305.17926, 2023. https://arxiv.org/abs/2305.17926, May 2024.

[110] Reynolds L, McDonell K. Prompt programming for large language models: Beyond the few-shot paradigm. arXiv: 2102.07350, 2021. https://arxiv.org/abs/2102.07350, May 2024.

[111] Dang H, Mecke L, Lehmann F, Goller S, Buschek D. How to prompt? Opportunities and challenges of zero- and few-shot learning for human-AI interaction in creative applications of generative models. arXiv: 2209.01390, 2022. https://arxiv.org/abs/2209.01390, May 2024.

[112] Zhang S Y, Dong L F, Li X Y, Zhang S, Sun X F, Wang S H, Li J W, Hu R Y, Zhang T W, Wu F, Wang G Y.

Instruction tuning for large language models: A survey. arXiv: 2308.10792, 2023. https://arxiv.org/abs/2307.04657, May 2024.

[113] Xu C, Sun Q F, Zheng K, Geng X B, Zhao P, Feng J Z, Tao C Y, Jiang D X. WizardLM: Empowering large language models to follow complex instructions. arXiv: 2304.12244, 2023. https://arxiv.org/abs/2304.12244, May 2024.

[114] Chung J, Kamar E, Amershi S. Increasing diversity while maintaining accuracy: Text data generation with large language models and human interventions. In *Proc. the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Jul. 2023, pp.575–593. DOI: 10.18653/v1/2023.acl-long.34.

[115] Howard J, Ruder S. Universal language model fine-tuning for text classification. arXiv: 1801.06146, 2018. https://arxiv.org/abs/1801.06146, May 2024.

[116] Mehrafarin H, Rajaee S, Pilehvar M T. On the importance of data size in probing fine-tuned models. arXiv: 2203.09627, 2022. https://arxiv.org/abs/2203.09627, May 2024.

[117] Chung H W, Hou L, Longpre S *et al.* Scaling instruction-finetuned language models. arXiv: 2210.11416, 2022. https://arxiv.org/abs/2210.11416, May 2024.

[118] Xue L T, Constant N, Roberts A, Kale M, Al-Rfou R, Siddhant A, Barua A, Raffel C. mT5: A massively multilingual pre-trained text-to-text transformer. arXiv: 2010.11934, 2021. https://arxiv.org/abs/2010.11934, May 2024.

[119] Lai V D, Ngo N T, Veyseh A P B, Man H, Dernoncourt F, Bui T, Nguyen T H. ChatGPT beyond English: Towards a comprehensive evaluation of large language models in multilingual learning. arXiv: 2304.05613, 2023. https://arxiv.org/abs/2304.05613, May 2024.

[120] Xu L, Zhang X W, Dong Q Q. CLUECorpus2020: A large-scale Chinese corpus for pre-training language model. arXiv: 2003.01355, 2020. https://arxiv.org/abs/2003.01355, May 2024.

[121] Gu Y, Tinn R, Cheng H, Lucas M, Usuyama N, Liu X D, Naumann T, Gao J F, Poon H. Domain-specific language model pretraining for biomedical natural language processing. *ACM Trans. Computing for Healthcare*, 2022, 3(1): 2. DOI: 10.1145/3458754.

[122] Ren X Z, Zhou P Y, Meng X F *et al.* PanGu-Σ: Towards trillion parameter language model with sparse heterogeneous computing. arXiv: 2303.10845, 2023. https://arxiv.org/abs/2303.10845, May 2024.

[123] Zhang R R, Han J M, Liu C, Gao P, Zhou A J, Hu X F, Yan S, Lu P, Li H S, Qiao Y. LLaMA-Adapter: Efficient fine-tuning of language models with zero-init attention. arXiv: 2303.16199, 2023. https://arxiv.org/abs/2303.16199, May 2024.

[124] Jiao W X, Huang J T, Wang W X, He Z W, Liang T, Wang X, Shi S M, Tu Z P. ParroT: Translating during chat using large language models tuned with human translation and feedback. arXiv: 2304.02426, 2023. https://arxiv.org/abs/2304.02426, May 2024.

[125] Xie Q Q, Han W G, Zhang X, Lai Y Z, Peng M, Lopez-Lira A, Huang J M. PIXIU: A large language model, instruction data and evaluation benchmark for finance. arXiv: 2306.05443,2023.https://arxiv.org/abs/2306.05443, May 2024.

[126] Wang H C, Liu C, Xi N W, Qiang Z W, Zhao S D, Qin B, Liu T. HuaTuo: Tuning LLaMA model with Chinese medical knowledge. arXiv: 2304.06975, 2023. https://arxiv.org/abs/2304.06975, May 2024.

[127] Bowman S R. Eight things to know about large language models. arXiv: 2304.00612, 2023. https://arxiv.org/abs/2304.00612, May 2024.

[128] Wang Y Z, Ivison H, Dasigi P, Hessel J, Khot T, Chandu K R, Wadden D, MacMillan K, Smith N A, Beltagy I, Hajishirzi H. How far can camels go? Exploring the state of instruction tuning on open resources. arXiv: 2306.04751, 2023. https://arxiv.org/abs/2306.04751, May 2024.

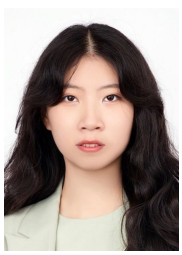
[129] Shi X M, Xu J, Ding J R, Pang J L, Liu S C, Luo S Q, Peng X W, Lu L, Yang H H, Hu M T, Ruan T, Zhang S T. LLM-Mini-CEX: Automatic evaluation of large language model for diagnostic conversation. arXiv: 2308.07635, 2023. https://arxiv.org/abs/2308.07635, May 2024.

[130] Ganguli D, Lovitt L, Kernion J *et al.* Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned. arXiv: 2209.07858, 2022. http://export.arxiv.org/abs/2209.07858v2, May 2024.

[131] Rillig M C, Ågerstrand M, Bi M, Gould K A, Sauerland U. Risks and benefits of large language models for the environment. *Environmental Science & Technology*, 2023, 57(9): 3464–3466. DOI: 10.1021/acs.est.3c01106.

[132] Anand Y, Nussbaum Z, Duderstadt B, Schmidt B, Mulyar A. GPT4All: Training an assistant-style chatbot with large scale data distillation from GPT-3.5-Turbo. Technical Report, 2023. https://s3.amazonaws.com/static.nomic.ai/gpt4all/2023_GPT4All_Technical_Report.pdf, May 2024.

[133] Li C Y, Wong C, Zhang S, Usuyama N, Liu H T, Yang J W, Naumann T, Poon H, Gao J F. LLaVA-Med: Training a large language-and-vision assistant for biomedicine in one day. arXiv: 2306.00890, 2023. https://arxiv.org/abs/2306.00890, May 2024.



Fei Du received her B.S. degrees in mathematical science and data science from University of California, Santa Barbara. She is currently a master student at Columbia University in New York City, and has interned in the National Key Laboratory of Data Space Technology and System and the Advanced Institute of Big Data, Beijing. Her research interests include machine learning, NLP, deep learning, and statistical data analysis.

Xin-Jian Ma received his Ph.D. degree in information security from the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, in 2017. He is currently an associate researcher at Advanced Institute of Big Data, Beijing. His research interests include big data, distributed system, and information security.

Jing-Ru Yang received her Ph.D. degree in computer science and technology from Renmin University of China, Beijing. She completed her postdoctoral research with funding from the Boya Program at Peking University, Beijing, and is currently a research associate of the National Key Laboratory of Dataspace Technology and System, Beijing. Her research focuses on data governance technology and systems.

Yi Liu received his Ph.D. degree in software engineering from Peking University, Beijing, in 2019. He is currently an associate researcher at Advanced Institute of Big Data, Beijing. His research interests include serverless computing and service computing.

Chao-Ran Luo received his Ph.D. degree in software engineering from Peking University, Beijing. He is currently a research associate at Advanced Institute of Big Data, Beijing. His research focuses on data space, internet of data, and digital object architecture.

Xue-Bin Wang received his Ph.D. degree in computer science from National University of Defence Technology, Changsha, in 2007. He is currently a senior engineer at Advanced Institute of Big Data, Beijing. His research interests are big data and artificial intelligence.

Hai-Ou Jiang received her Ph.D. degree in computer science from Beijing University of Posts and Telecommunications, Beijing, in 2017. She is currently a research associate at Advanced Institute of Big Data, Beijing. Her research interests include cloud computing, big data, and machine learning.

Xiang Jing received his Ph.D. degree in information security from the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, in 2016. He finished his postdoctoral program in computer theory from Software Engineering Institute at Peking University, Beijing. He is currently a research associate in the School of Software and Microelectronics at Peking University, Beijing. His research focuses on operating system, big data, blockchain technology, and industrial internet.