

Ziya2: Data-centric Learning is All LLMs Need

Ruyi Gan^{♥♠} Ziwei Wu[♥] Renliang Sun[♥] Junyu Lu[♥] Xiaojun Wu[♥]
 Dixiang Zhang[♥] Kunhao Pan[♥] Ping Yang^{♥*} Qi Yang^{♥*}
 Jiaxing Zhang[♥] Yan Song[♠]

[♥]International Digital Economy Academy [♠]University of Science and Technology of China
 {ganruiyi, wuziwei, sunrenliang, lujunyu, zhangjiaxing}@idea.edu.cn
 clkson@gmail.com

Abstract

Various large language models (LLMs) have been proposed in recent years, including closed- and open-source ones, continually setting new records on multiple benchmarks. However, the development of LLMs still faces several issues, such as high cost of training models from scratch, and continual pre-training leading to catastrophic forgetting, etc. Although many such issues are addressed along the line of research on LLMs, an important yet practical limitation is that many studies overly pursue enlarging model sizes without comprehensively analyzing and optimizing the use of pre-training data in their learning process, as well as appropriate organization and leveraging of such data in training LLMs under cost-effective settings. In this work, we propose Ziya2, a model with 13 billion parameters adopting LLaMA2 as the foundation model, and further pre-trained on 700 billion tokens, where we focus on pre-training techniques and use data-centric optimization to enhance the learning process of Ziya2 on different stages. Experiments show that Ziya2 significantly outperforms other models in multiple benchmarks especially with promising results compared to representative open-source ones.¹

1 Introduction

LLMs have achieved great success in the field of artificial intelligence (AI), especially natural language processing (NLP), over the past few years. Generally, LLMs are pre-trained on large amounts of text and show promising performance in a variety of NLP tasks without requiring intensive task-specific tuning on huge amounts of labeled data (Devlin et al., 2019; Raffel et al., 2020; Joshi et al., 2020; Qin et al., 2021; Wang et al., 2022; Lu et al., 2022; Tian et al., 2023; Ping et al., 2023; Huang

et al., 2023a). Among all LLMs, representative ones include GPT-3 (Brown et al., 2020) and its successors ChatGPT (OpenAI, 2022), GPT-4 (OpenAI, 2023), and PaLM-2 (Anil et al., 2023), demonstrate strong adaptability and applicability. However, owing to the fact that the aforementioned LLMs are developed in rather restricted environments, the lack of access to developers’ source code and parameters becomes a barrier for many following researchers and developers in continuing LLM research based on existing well-performed models. This paradox leads to the phenomenon that many researchers turn to train open-source alternatives, such as LLaMA2 (Touvron et al., 2023b) and Falcon (Penedo et al., 2023), etc., since the open-source LLMs provide a foundation for further learning and improvement on the shoulders of successful predecessors, which promotes transparency and accountability in following AI studies.

Yet, besides a variety of benefits that have been brought along with LLMs, especially the open-source counterparts, their development is currently facing the following three significant challenges. The first one is the high cost of pre-training models from scratch, making LLM learning a cost process, so that continual training (or pre-training) is potentially cost-effective when starting from a foundation LLM. The second one is that open-source LLMs often do not come with open-source data, leading to discrepancies in data distribution, implicitly resulting in catastrophic forgetting when mixed with different data in later usage. The third one is that many studies on LLMs prioritize increasing model size so as to maximize their benefits from various aspects, and they thus often overlook the influence of training data quality on model performance, which has been demonstrated to be significant in many NLP tasks (Liu et al., 2019a; Brown et al., 2020; Black et al., 2022; Anil et al., 2023; Touvron et al., 2023b). The pursuit of enlarging model sizes necessitates more extensive GPU con-

^{*}They contributed to this work when they worked in IDEA.

¹Ziya2 (Base) is released at <https://huggingface.co/IDEA-CCNL/Ziya2-13B-Base> and <https://modelscope.cn/models/Fengshenbang/Ziya2-13B-Base/summary>

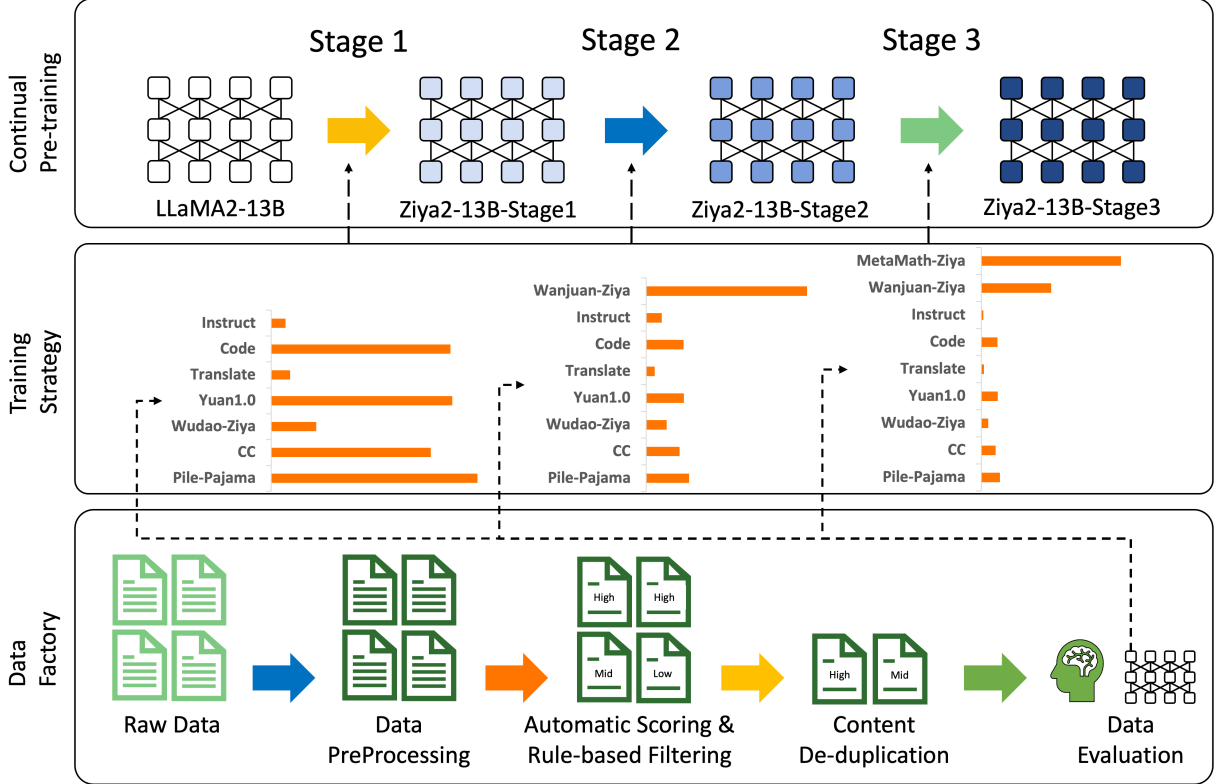


Figure 1: The overall data-centric process for learning Ziya2, where the pipeline to obtain high-quality data, the training strategies, and the three-stage training process are presented in the bottom, middle, and top part of the figure, respectively. Notably, the training strategies for different stages are illustrated by comparisons of data distributions.

sumption, accordingly increasing training costs as well. Therefore, such challenges ferment expectations of investigating data quality and its impact on the LLMs and research conducting on better leveraging data for LLM learning at different stages.

In this work, we focus on the technique of continual pre-training and understanding the intricate relationship between data and model performance. We delve into an in-depth exploration of how the high-quality pre-training data enhance the performance of an LLM, while keeping its size and structure essentially unchanged. In doing so, we build Ziya2 by using LLaMA2-13B (Touvron et al., 2023b) as the foundational model and continue pre-training it on approximately 700 billion tokens in Chinese and English. Specifically, a three-stage training process has been adopted to leverage both general and domain-specific corpora to enhance Ziya2’s bilingual generation capabilities, where the first stage trains Ziya2 with huge high-quality data, including Chinese and English languages; the second stage uses supervised data to optimize the LLM; and the third stage mainly focuses on training Ziya2 with mathematical data. The entire learning process culminates in the creation of the Ziya2 model with 13

billion parameters, as that shown in Figure 1.

The evaluation of Ziya2 is performed on several representative benchmarks including multidisciplinary datasets in Chinese and English, as well as that specifically designed to test mathematical and programming abilities. Compared to LLaMA2 (Touvron et al., 2023b), Ziya2 shows its superiority significantly across different benchmarks. Particularly, using LLaMA2 as a standard LLM assessment, Ziya2 improves it by **10% on MMLU** (Hendrycks et al., 2020), **61% on CMMLU** (Li et al., 2023), and **68% on C-Eval** (Huang et al., 2023b), respectively. Especially in terms of mathematical benchmarks, Ziya2 improves LLaMA2 by **138% on GSM8K** (Cobbe et al., 2021) and **120% on MATH** (Hendrycks et al., 2021), and for programming benchmarks, Ziya2 improves it by **89% on HumanEval** (Chen et al., 2021). For further illustration of the effectiveness of Ziya2, we also select several representative open-source models of similar size for comparison. **It is shown that Ziya2 achieves outstanding performance on multidisciplinary datasets²**, where, especially, it surpasses all open-source models used for com-

²Detailed results are reported in Table 3.

parison in mathematics and programming skills. Notably, Ziya2’s performance on Chinese tasks also surpasses the GPT-3.5-turbo³. These results demonstrate that high-quality data and appropriate continual pre-training are able to significantly improve LLMs’ performance without significantly changing model size and affecting their stability.

2 The Approach

The overview of our learning process for Ziya2 is illustrated in Figure 1, where it commences with the establishment of a data pipeline capable of continuous cleansing and evaluation of extensive web-based datasets, ensuring the acquisition of large-scale, high-quality data for the training of Ziya2. Subsequently, resource conservation and the exploration of effective continual pre-training strategies are facilitated by leveraging LLaMA2-13B as the foundation for extensive further pre-training, culminating in the completing of Ziya2. This process still necessitates large-scale domain-specific data to augment particular capabilities. Therefore, a data-centric, three-stage continual pre-training strategy is employed to endow Ziya2 with superior capabilities in bilingual understanding, code generation, and mathematical reasoning compared to LLaMA2-13B, with details on data processing, model architecture and training optimization illustrated in following texts.

2.1 The Data Factory

Data serves as a pivotal cornerstone for LLMs, with its scale and quality directly influencing model performance. Based on this fact, we develop a proprietary data factory based on CCNet (Wenzek et al., 2019), as depicted in Figure 2, to facilitate a multitude of functions including data preprocessing, automatic scoring, rule-based filtering, content de-duplication, and data evaluation. This factory is specifically tailored in this work to meet the requirements of continual pre-training, data sampling and augmentation. In doing so, it is able to continuously process data from the Internet at a rate of 15GB per hour, ensuring a consistent influx of refined data to further enhance our LLM training process.⁴ Details are illustrated in following texts.

Data Preprocessing (DP) This is the first step in data processing, in which we conduct several

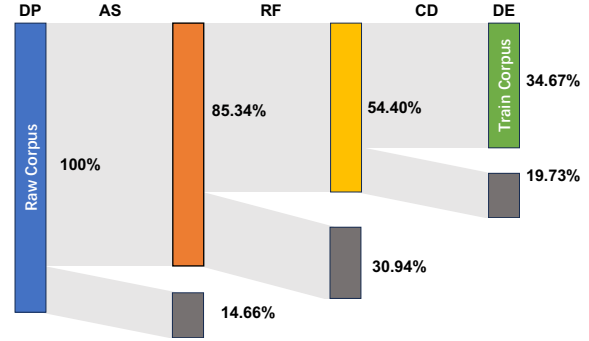


Figure 2: An overview of the proposed data-centric learning approach of Ziya2, which includes five components: data preprocessing (DP), automatic scoring (AS), rule-based filtering (RF), content de-duplication (CD), and data evaluation (DE). The gray and colored blocks indicate the proportion of data that is filtered out and retained relative to the original dataset, respectively.

basic operations. The first is language detection on the collected corpus and selection of only Chinese and English language data. Once the selection is done, we proceed to standardize the encoding of the corpus and transform all Chinese text into the simplified version. Then, we remove non-usable tokens such as non-visible control characters, special symbols, emojis, and improper punctuation marks.

Automatic Scoring (AS) Upon the resulted data from preprocessing, we perform an automatic quality control (filtering) by language model scoring. In doing so, we employ KenLM (Heafield, 2011) to train two language models from Chinese and English Wikipedia, respectively, and conduct perplexity (PPL) assessments on the input data. Subsequently, we select data based on their ranking from lower to higher PPL scores, with the top 30% data marking as high-quality and those in between 30% and 60% in their PPL ranking as medium quality.

Rule-based Filtering (RF) Given the abundance of sensitive and unsafe content on the Internet, which are easy to be memorized by LLMs, so that it is required to eliminate text with significant tendencies toward pornography, violence, politics, and advertising. Therefore, we design more than 30 filtering rules at three granular levels: document, paragraph, and sentence, and filter out texts by applying rules from large to small granular levels. At the document level, rules are principally designed around content length and format, while at the paragraph and sentence levels, the focus of rules shifts to the toxicity of the content. It is worth noting that, in the preliminary stages of rule design, we

³We use the model under its “gpt-3.5-turbo-0613” version.

⁴With our data factory, we accumulate a volume of more than 4.5TB of high-quality data in our internal repository.

Metric	Method	Level	Comments
Privacy Word Rate	Re Matching	1	Includes email addresses, phone numbers and ect.
Toxic Word Rate	Re Matching	1	Contains political, explicit, violent, or similar content.
Adv Word Rate	Re Matching	1	Checking for the presence of advertising keywords.
Webpage Funcword Rate	Re Matching	1	Detecting the presence of HTML tags.
Perplexity	LM	2	Whether the sentences in the article are coherent.
Informativeness	LM	2	Assessing the information content within the text.
Readability	Counts	2	Assessing text readability
Language	LM	3	Analyzing the distribution of languages.
Doc Length	Counts	3	Evaluating the distribution of text lengths.
Topic	Statistics	3	Detecting the topic distribution of the text.

(a) Machine Metrics

Metric	Method	Level	Comments
Incoherent	Human Check	1	Assessing text readability
Sensitive Topics	Human Check	1	Contains political, explicit, violent, or similar content.
Redundancy	Human Check	1	To detect if the text contains redundant information.

(b) Human Metrics

Table 1: These are the metrics we utilize for evaluating data quality. The **Method** denotes our detection method, **Re Matching** signifies the method involving regular expressions for counting. **LM** represents the utilization of a language model for predicting relevant metrics. **Counts** indicates the use of statistical methods for directly countable metrics. **Human Check** indicates manual spot-checking conducted by humans. **Level** indicates the degree of stringency we apply to the respective metrics. We consider it compliant if it does not exceed one in a thousand.

also conduct random sampling of the original text for manual evaluation. Rule iterations are then informed by feedback from human reviews, ensuring the precision of later filtration process.

Content De-duplication (CD) As mentioned in the following studies (Lee et al., 2021; Tirumala et al., 2023; Penedo et al., 2023), repetitive content does not significantly improve training and hurts training efficiency. We used Bloomfilter (Bloom, 1970) and Simhash (Charikar, 2002) to de-duplicate the text in the data through the following steps. First, we find that Common Crawl and other open-source datasets contain substantial duplication of web pages and thus use Bloomfilter⁵ to de-duplicate URLs, which significantly reduces the computational load required for subsequent content de-duplication. Second, our analysis shows that many remaining web pages share similar content, where the main differences among them are special characters such as punctuation marks and emoticons. Thus, we perform a round of precise de-duplication on these web pages. Third, we employ SimHash for the fuzzy de-duplication of textual content for the remaining data. Although this process may incorrectly exclude some high-quality

data, our manual sampling evaluation of the de-duplicated data indicates that this loss is acceptable in order to pursue higher efficiency. Moreover, to facilitate efficient content de-duplication for new data collected by our data factory, we use caching and bucketing techniques, where new data do not necessitate redundancy checks against all existing data, which optimizes the de-duplication process.

Data Evaluation (DE) After data cleaning, we evaluate the data quality with the machine and human metrics illustrated in Table 1. For machine metrics, we randomly select 1% of the processed data and run scripts to compute the machine metrics. For human metrics, we randomly sample 1,000 instances and manually evaluate the extracted corpus accordingly. Based on the machine and human metrics, we determine whether an example qualifies our standard for data quality. Afterwards, we compute the rate of the unqualified examples over all evaluated instances. If the rate is lower than a threshold, it proves the data meets our requirement, so we use them as a part of the training corpus. If the rate is higher than the threshold, which means the data does not meet our standard, we improve the processes of automatic scoring, rule-based filtering, and content de-duplication, which

⁵We use our own implementation of Bloomfilter.

Pre-training Stage	Dataset	Language	Size	Doc #	Token #	Sampling
Stage 1/2/3	Pile-Pajama	en	800GB	94M	220B	50%
	CC	en	600GB	81M	170B	64%
	Wudao-Ziya	zh	156GB	51M	48B	100%
	Yuan1.0	zh	54GB	260M	193B	100%
	Translate	multi	3GB	12M	1.5B	100%
	Code	multi	480GB	124M	191B	100%
	Instruct	multi	1.6GB	0.9M	0.8B	100%
Stage 2/3	Wanjuan-Ziya	zh	76GB	16M	29B	100%
Stage 3	MetaMath-Ziya	en	0.3GB	0.4M	0.1B	100%

Table 2: The information of high-quality pre-training datasets used to train Ziya2. The “pre-training stage” illustrates the stages that the datasets are used; “en”, “zh”, “multi” mean the language of the datasets are English, Chinese, and multi-lingual, respectively; “Sampling” refers to the ratio of data sampled from the original dataset.

are then utilized to process data.

The Resulted High-quality Data By cleaning datasets from Common Crawl⁶, Wudao (Yuan et al., 2021), Yuan1.0 (Wu et al., 2021), Pile (Gao et al., 2020), Redpajama (Computer, 2023), Wanjuan (He et al., 2023), MetaMath (Yu et al., 2023) and our own collected data, which included code and book-related content, we sample a total of 4.5TB of high-quality data from the initial 13TB of open-source data. Table 2 lists primary information (i.e., the language, size, number of tokens and documents, and the ratio of data sampled from datasets) about the processed data, whose details are illustrated as follows. **Pile-Pajama** is a de-duplicated fusion of the Pile and Redpajama datasets after removing Common Crawl data. **CC** is a de-duplicated fusion of data from Pile and Redpajama that originated from Common Crawl. **Wudao-Ziya** is a dataset that combines our collected data with the Wudao dataset. **Yuan1.0** is an open-source dataset provided by Inspur Technology, and we filter the raw data using our cleaning rules. **Translate** is the multi-lingual translation dataset we collect. **Code** is the code data we collect from GitHub, which includes multiple programming languages such as C, C++, and Python. We add the program language type before the code and change it to a format that the Markdown syntax is able to recognize. In this way, the model we train is able to generate formatted code. **Instruct** is a dataset constructed from instructions that we collect. **Wanjuan-Ziya** is a dataset that combines high-quality data from the Wanjuan dataset, as well as math-related data we collect ourselves. **MetaMath-Ziya** is a dataset derived

from the Huawei’s open-source MetaMath dataset after data augmentation. We construct some Chinese and English prompts for **Instruct**, **Wanjuan-Ziya**, and **MetaMath-Ziya** datasets, such as “*Q-A*”, “*question-answer*”, “*problem-solution*”, etc.⁷

2.2 The Architecture of Ziya2

The architecture of Ziya2 is based on LLaMA2, where we propose to improve the quality of input data processing, token and hidden representations through enhancing tokenizer, positional embedding, as well as layer normalization and attention, respectively, so as to facilitate Chinese text processing, adapt to the changes in text length and data distribution, and improve its efficiency and stability in pre-training. The details of the aforementioned improvements are illustrated as follows.

Tokenizer In order to maximize the preservation of semantic meaning in the text and provide better support for Chinese, we adopt a BPE (Byte-Pair Encoding) tokenizer (Sennrich et al., 2015). For the vocabulary of the tokenizer, we reuse over 600 Chinese characters originally used in LLaMA and add extra 7,400 commonly used Chinese characters, which include both simplified and traditional Chinese characters as well as Chinese punctuation marks. The reason for adding extra Chinese characters is the following. The original LLaMA2 vocabulary with BPE tokenizer is not highly efficient for Chinese, primarily due to how computers handle character encoding. For instance, in most cases, one UTF-8 Chinese character is encoded into 2-4 tokens using BPE encoding. After adding Chinese characters to the vocabulary, testing on a 10GB

⁶<https://commoncrawl.org/>

⁷Some examples training data are shown in Appendix A.

Chinese corpus shows an efficiency improvement of approximately 34% in Chinese encoding and decoding compared to the original tokenizer.

Positional Embedding During the continual pre-training, we observe that there is a divergence in the distribution of text lengths between our continued training dataset and the original LLaMA2 dataset, necessitating an adaptation of the position embedding to different data distributions. Meanwhile, considering downstream tasks involving lengthy texts, the scalability of position embedding is of significant importance. LLaMA2 employs rotary position encoding (Su et al., 2021), which, through the mechanism of absolute position encoding, accomplishes relative position encoding. To avoid the overflow issues associated with mixed precision, we implement rotary position encoding using FP32 precision, thereby accommodating the variation in data length distribution in continual training.

Layer Normalization and Attention During the continual pre-training process, we find that due to divergences in the data distribution between our continued training dataset and the original LLaMA2 dataset, the direct implementation of mixed precision training in layer normalization and attention leads to precision overflow, which results in training instability. To maintain the efficiency and stability of the model training, we improve layer normalization and attention in LLaMA2. Specifically, for layer normalization, we utilize an APEX⁸ RMSNorm (Zhang and Sennrich, 2019) implementation, which also operates under FP32 precision training. For attention, we employ a fused operator to replace the original scaling, mask, and softmax operators within the attention module, thereby expediting the computation of attention. Furthermore, to prevent overflow during mixed precision training in the softmax operator, it is trained using FP32 precision. Through these structural improvements, the training process is better adapted to the instability brought about by changes in data distribution, thus enhancing training efficiency.

2.3 Continual Pre-training

We continually pre-train LLaMA2 on datasets with three-stage pre-training to benefit from high-quality data in training Ziya2 model. The following text illustrates the pre-training process, which covers

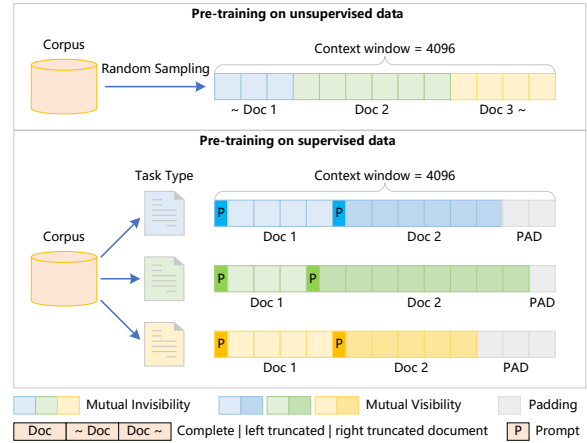


Figure 3: The process of constructing unsupervised and supervised pre-training data in the three stages.

model initialization, training strategies, optimizer, training efficiency, and training stability.

Initialization Considering the vocabulary with 8,000 Chinese characters, during continual pre-training, each Chinese character represents a new token. For this token, in the continual pre-training of Ziya-13B, we employ random initialization. Given that LLaMA2 already learns some information related to Chinese and in LLaMA2, one Chinese character corresponds to 2-4 tokens; we take the weighted average of the embeddings of these tokens as the embedding for the Chinese character in Ziya2. We then proceed with further pre-training.

Training Strategy During continual pre-training, a critical challenge is catastrophic forgetting. The goal is to enhance LLaMA2’s Chinese and coding capabilities without compromising its proficiency in English, which is also Ziya2’s objective. To achieve this, we conduct continual pre-training in three stages, where the details of data concatenation for these stages are illustrated in Figure 3. The following texts present the details of the three stages.

In the first stage, as shown in Table 2, we sample English data close to LLaMA2’s original distribution. We also sample Chinese data from Wudao-Ziya and Yuan 1.0, as well as data from Code, which together form 650B unsupervised data for continual pre-training. We thoroughly shuffle these datasets, concatenate different data segments into a 4,096 context as a training example, and use an attention mask to prevent different data segments from influencing each other. This design maximizes training efficiency and enables Ziya2 to possess powerful Chinese and coding knowledge.

In the second stage, we enhance Ziya2’s perfor-

⁸<https://github.com/NVIDIA/apex>

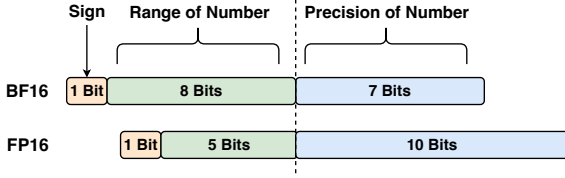


Figure 4: An illustration of the representations of floating numbers by BF16 and FP16.

mance on downstream tasks by adding supervised data with Chinese and English instructions. Different from fine-tuning with instructions, we still employ the LLM’s next token prediction training method. Instead of randomly combining data as in the first stage, we concatenate the same type of instruct data into a 4,096 context as a training example, where the rest of the positions are filled by special pad tokens. To retain the knowledge already acquired by Ziya2, we also sample unsupervised Chinese and English data in the same proportion as the instruct data for continual training.

In the third stage, we incorporate supervised data related to inference, such as Wanjian-Ziya and MetaMath-Ziya, using the same composition approach as in the second stage. After the third stage, Ziya2 shows significant improvement in mathematical reasoning abilities and exhibits noticeable enhancement in coding capabilities. This indicates that mathematical reasoning skills are vital for tasks (e.g., coding) that require reasoning.

Optimizer We train the model using the AdamW (Loshchilov and Hutter, 2017) optimizer with the following hyperparameters: $\beta_1 = 0.9$, $\beta_2 = 0.95$. In our findings, due to the incorporation of additional Chinese and code data in our dataset compared to LLaMA2, there exists a disparity in the overall data distribution. As a result, a more extended warmup is beneficial for continual pre-training. Consequently, instead of the 0.4% warmup ratio utilized in LLaMA2, we adopt a warmup ratio of 1%. Followed by a cosine learning rate decay schedule, reaching a final learning rate of $1e^{-5}$. Additionally, we implement a weight decay of 0.1 and gradient clipping set at 1.0.

Training Efficiency We utilize Megatron (Shoeybi et al., 2019) and DeepSpeed as foundational frameworks. Megatron ensures distributed training of large-scale models through data parallelism, tensor parallelism, and pipeline parallelism. Concurrently, the ZeRO (Rajbhandari et al., 2020) technology from DeepSpeed is

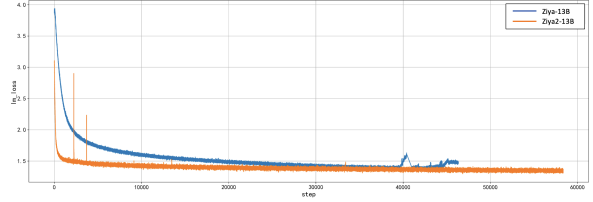


Figure 5: The pre-training loss of Ziya2-13B and Ziya2-13B with respect to the number of training steps.

employed to optimize memory savings. Building on this, we have implemented adaptations for the LLaMA2 model structure and incorporated advanced techniques such as flash-attention (Dao et al., 2022) and fused-softmax to enhance training efficiency. This results in an industry-leading efficiency of 163.0 TFLOPS per GPU per second and thus guarantees efficient training.

Training Stability Ziya-13B adopts FP16 mixed-precision training (Micikevicius et al., 2017) and frequently encounters loss spike issues in the later stages of training. Upon analysis, we identify that the limited numerical range of FP16 leads to overflow problems, especially in operations such as softmax. As illustrated in Figure 4, in comparison to FP16, BF16 offers superior precision. Hence, for the continual pre-training of Ziya2-13B, we opt for BF16 mixed-precision training. Simultaneously, we rectify some underlying bugs in DeepSpeed to ensure the stability of training under BF16 precision, addressing the loss spike challenges. This ensures convergence even when training continues for 700B tokens, as shown in Figure 5.

3 Evaluation

We evaluate Ziya2 on benchmark datasets and compare its performance with state-of-the-art LLMs, where the results and further analysis of data efficiency demonstrate the effectiveness of Ziya2 with the proposed data-centric learning process. The details are illustrated in the following text.

3.1 Datasets

Following existing studies (Zeng et al., 2022; Touvron et al., 2023b; Yang et al., 2023), we evaluate Ziya2 on six benchmark datasets using OpenCompass (Contributors, 2023), where we perform 5-shot in experiments and use the official evaluation scripts to compute the accuracy. The details of the datasets are illustrated as follows.

- **MMLU** (Massive Multitask Language Understanding) (Hendrycks et al., 2020) provides a

Source	Model	MMLU	CMMLU	C-Eval	GSM8K	MATH	HumanEval
Closed-source	GPT-4	83.93	70.33	68.40	89.99	40.20	69.51
	GPT-3.5-turbo	68.54	54.06	51.10	57.77	13.96	52.44
Open-source	ChatGLM2-6B	47.86	49.30	51.70	28.89	6.40	9.15
	Falcon-7B	26.03	25.66	24.23	5.46	1.68	-
	Vicuna-13B	52.00	36.28	32.80	28.13	4.36	16.46
	XVERSE-13B	55.21	58.44	53.70	18.20	2.18	15.85
	WeMix-13B	59.70	42.60	42.70	45.20	6.90	24.40
	Baichuan2-13B	59.17	61.97	58.10	52.77	10.08	17.07
	LLaMA2-13B	55.74	37.90	34.99	28.81	4.98	15.24
Ours	Ziya-13B	43.88	31.09	28.97	17.97	3.08	14.02
	Ziya2-13B	61.36	60.95	58.84	68.46	10.98	28.87

Table 3: Comparison of Ziya2 with other closed-source and open-source LLMs on six benchmark datasets for LLM evaluation, where the **boldface** indicates the best-performing result over all open-source LLMs.

comprehensive evaluation of models in both zero-shot and few-shot settings, spanning across a wide range of 57 subjects. The unique aspect of MMLU is that it tests the models’ world knowledge and problem-solving capabilities.

- **CMMLU** (Chinese Massive Multitask Language Understanding) (Li et al., 2023) is an extensive evaluation suite tailored to measure the advanced knowledge and reasoning skills of LLMs in the context of the Chinese language and culture. It contains a broad spectrum of 67 topics, ranging from basic to advanced professional levels.
- **C-Eval** (Chinese Evaluation) (Huang et al., 2023b) is a thorough evaluation suite specifically designed for foundational models in Chinese. It is composed of 13,948 multiple-choice questions that cover a wide array of 52 different disciplines and span across four levels of difficulty.
- **GSM8K** (Grade School Math 8K) (Cobbe et al., 2021) is a collection of 8,500 high-quality math word problems created by human writers. The objective of this dataset is to facilitate the task of question answering on fundamental problems that necessitate reasoning through multiple steps.
- **MATH** (Hendrycks et al., 2021) aggregates 12,500 intricate competition mathematics problems. A unique feature of this dataset is that each problem comes with a comprehensive step-by-step solution. These detailed solutions serve as a valuable resource for teaching models to generate derivation processes and explanations.
- **HumanEval** (Chen et al., 2021) is a meticulously constructed set of 164 programming challenges. This dataset serves as a benchmark for evalu-

ating the ability of a system to generate functionally correct programs based on provided doc strings. The challenges encompass a range of topics, including language comprehension, algorithmic problems, and basic mathematics.

3.2 Baselines

We choose several open-source models with similar sizes to Ziya2 as baselines.

- **ChatGLM2-6B** (Zeng et al., 2022) is developed by Tsinghua University and Zhipu AI. The model is the second generation of the bilingual dialogue model ChatGLM-6B.
- **Falcon-7B** (Penedo et al., 2023) is a causal decoder-only model developed by Technology Innovation Institute (TII). It’s trained on 1,500 billion tokens of RefinedWeb.
- **Vicuna-13B** (Chiang et al., 2023) is an open-source LLM developed by the Language Model Systems (LMSYS) organization and is fine-tuned from LLaMA on user-shared conversations between humans and ChatGPT.
- **Baichuan2-13B** (Yang et al., 2023) is developed by Baichuan Intelligent Technology and is trained with 2.6 trillion tokens.
- **XVERSE-13B**⁹ is developed by XVERSE Technology and is trained with 1.4 trillion tokens.
- **WeMix-13B**¹⁰ is developed by Shanghai AI lab and is built on LLaMA2-Accessory.
- **LLaMA2-13B** (Touvron et al., 2023b) is Meta’s open-source large language model. It is designed

⁹<https://github.com/xverse-ai/XVERSE-13B>.

¹⁰<https://github.com/Alpha-VLLM/WeMix-LLM>.

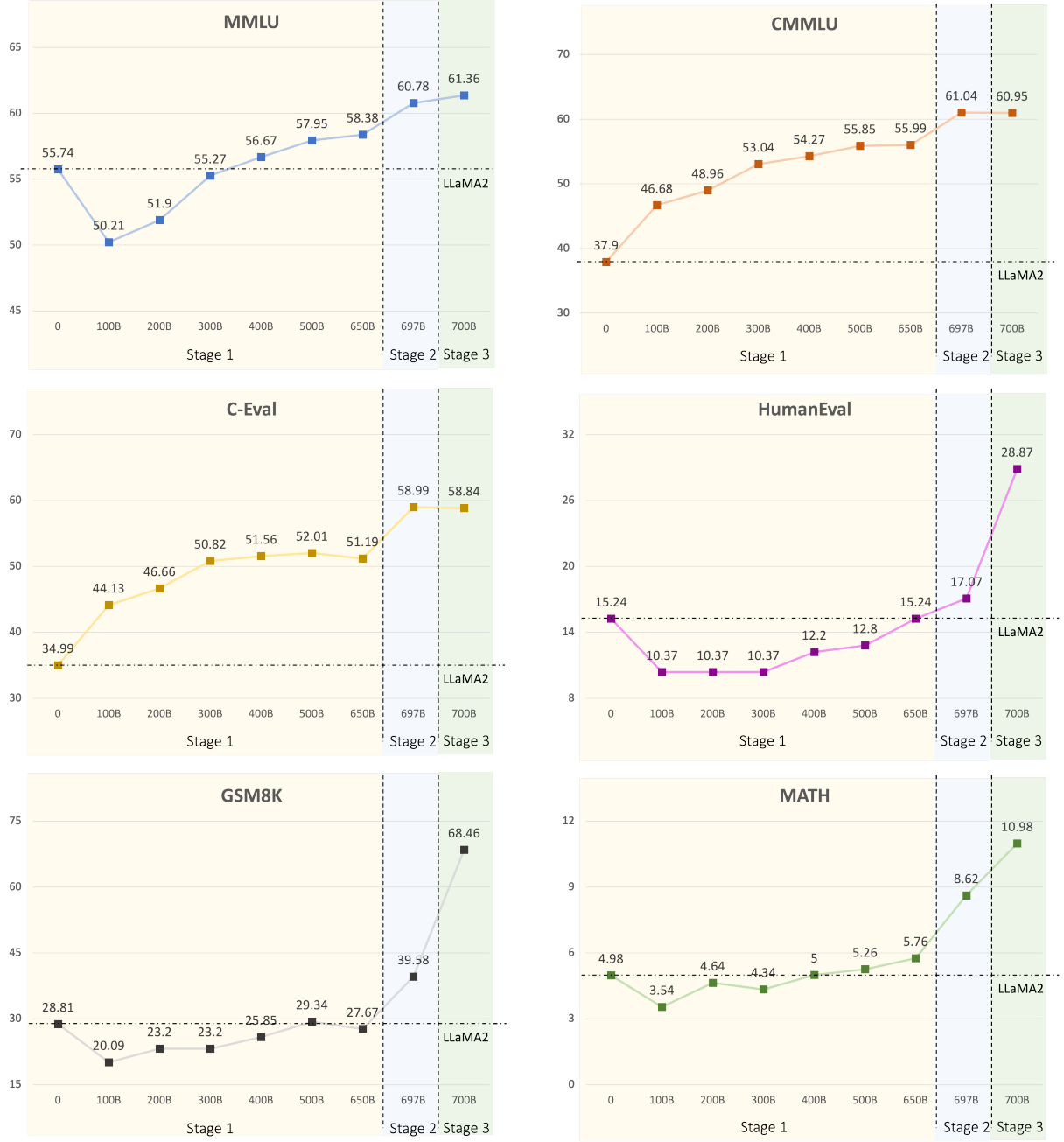


Figure 6: The performance (i.e., accuracy) of Ziya2 on the six benchmark datasets with respect to the three training stages. The training process in different stages is represented by the number of tokens used to train Ziya2 at a particular training step. The performance of LLaMA2 on the datasets is illustrated by dashed lines for reference.

for dialogue scenarios and is freely available for research and commercial use.

- **Ziya-13B** (Zhang et al., 2022) is continue pre-trained on the LLaMA-13B model and performs well on many downstream tasks.

We also choose closed-source models GPT-3.5-turbo and GPT-4 (OpenAI, 2023) as references.

- **GPT-3.5-turbo** is a high-performance variant of GPT-3 and it is proficient in text completion.
- **GPT-4** (OpenAI, 2023) is the latest state-of-

the-art LLM developed by OpenAI. It exhibits human-level performance on various professional and academic benchmark datasets.

3.3 Benchmark Results

The results of different LLMs on benchmark evaluations are shown in Table 3 with several observations.¹¹ First, Ziya2 significantly outperforms LLaMA2 and other representative open-source

¹¹In Appendix B, we present examples produced by Ziya2 across three major domains: Chinese, English, and code.

LLMs on all general Chinese and English datasets (i.e., C-Eval and MMLU). Remarkably, Ziya2’s Chinese proficiency is even superior than that of GPT-3.5-turbo. These results underscore the efficacy of our continual pre-training method, which not only enhances LLaMA2’s English capabilities and mitigates catastrophic forgetting but also substantially boosts its performance on Chinese tasks, which LLaMA2 does not focus on. Second, for specific downstream tasks, Ziya2 demonstrates remarkable mathematical and coding capabilities, where it outperforms all open-source LLMs, especially LLaMA2, and obtains better performance than GPT-3.5-turbo on GSM8K. In technical measuring, Ziya2 outperforms LLaMA2 by 40%, 6%, and 13% on GSM8K, MATH, and HumanEval datasets, respectively. These results indicate that continual pre-training not only enhances the performance of LLMs on general tasks, but also consistently improves their performance in specific fields.

3.4 Data Efficiency

To investigate data efficiency in training Ziya2, we present the performance (i.e., accuracy) of Ziya2 on six datasets with respect to the three training stages in Figure 6, where the results from LLaMA2 is represented by horizontal dashed lines for comparison. In doing so, we use the number of tokens that have been used to train Ziya2 at a particular training step to represent particular training periods, so as to better illustrate the effect of data on LLM performance. The following description presents the observations from the three training stages, which emphasize the contribution of high-quality data to the promising performance of Ziya2.

During the first training stage, initially, the performance of Ziya2 on MMLU degrades owing to the inclusion of a large amount of Chinese corpus in the training data that different from the setting of LLaMA2. With training step increases, Ziya2 learns from a broader view from more data, which enhances its capabilities in both Chinese and English text processing. Particularly, in Chinese tasks that LLaMA2 is not optimized for, new data significantly improves Ziya2’s performance on CMMLU and C-Eval benchmarks. In the meantime, a modest enhancement is synchronously observed in Ziya2’s mathematical and programming abilities.

In the second stage of continual pre-training, Ziya2 exhibits a more substantial enhancement on six benchmarks relative to the first stage, with

notable advancements on C-Eval, GSM8K, and MATH. These results underscore the greater contribution of supervised data over unsupervised data to LLM pre-training. Therefore, employing supervised data for pre-training is able to reduce the number of training steps, thereby economizing costs.

In the third stage, we observe that using the MetaMath dataset for training significantly enhances the performance of Ziya2 on GSM8K and MATH, while maintaining its performance on general tasks. On one hand, the experimental results prove that data augmentation specific to a particular dataset is able to significantly boost the model’s performance on that dataset. On the other hand, such an “effortless” enhancement in model performance may not necessarily be beneficial, as the model might merely learn the format of the problems, rather than genuinely improving its mathematical capabilities.

4 Related Work

In the past decade, text representation serves as the fundamental of NLP and its quality determines how well a model is able to perform on downstream tasks. Along the road of text representation research, pre-trained word embeddings and LLMs have demonstrated their effectiveness in providing high-quality text representations and thus achieved outstanding performance on many NLP tasks for years (Mikolov et al., 2013; Pennington et al., 2014; Song et al., 2017, 2018; Radford et al., 2019; Devlin et al., 2019; Liu et al., 2019b; Diao et al., 2020; Song et al., 2021; Tian et al., 2022; Yang et al., 2022). Among LLMs, decoder-only ones from GPT-3 (Brown et al., 2020) represents a significant journey in AI, since it demonstrates impressive zero- and few-shot learning performance by using remarkably large model size (i.e., 175B parameters). InstructGPT (Ouyang et al., 2022) further improves GPT-3 by supervised fine-tuning and reinforcement learning from human feedback (RLHF), where human-annotated data is used to train the LLM. Then, ChatGPT (OpenAI, 2022) is released as a general conversational model attracting much attention from the public since its noticeable capabilities in processing text. Later, GPT-4 (OpenAI, 2023) presents huge improvements over its predecessors and provides multimodal ability that able to process images and texts whenever needed.

In order to better utilize and study LLMs, people are committed to developing open-source models since GPT-3, instructGPT, ChatGPT, and GPT-4

are all not publicly available. LLaMA (Touvron et al., 2023a) and LLaMA2 (Touvron et al., 2023b) are two representative publicly available LLMs that have been used in many downstream tasks. LLaMA2 is trained mainly on English data with more tokens applied than LLaMA, and extends to generate longer text. In response to the subpar performance of open-source models such as LLaMA series, many researchers propose LLMs for Chinese text processing. For instance, Chinese-Vicuna (Fan et al., 2023) is a low-cost Chinese dialogue model based on LLaMA. ChatGLM2 (Zeng et al., 2022) is another LLM that supports dialogue in both Chinese and English, which is specifically optimized for Chinese dialogue tasks, achieving excellent results in Chinese question-answering and dialogue tasks. Baichuan2 (Yang et al., 2023) is an LLM that supports Chinese and has achieved commendable results on multi-disciplinary benchmarks in both Chinese and English. Compared with existing LLMs for Chinese, Ziya2 aligns with the model structure of LLaMA2, facilitating seamless integration with the established LLaMA ecosystem. This alignment ensures the adaptivity of Ziya2 on both academic and industrial purposes. In contrast to the Chinese LLMs that are trained extensively from scratch, Ziya2 achieves the same objectives of enhanced Chinese language comprehension, code generation, and mathematical reasoning through significant effective continual pre-training with data-centric optimizations. Our approach not only substantiates the efficacy of continual pre-training but also showcases its efficiency in augmenting the capabilities of existing models.

5 Conclusion

In this paper, we propose Ziya2, an open-source LLM with 13 billion parameters for Chinese and English text processing. Ziya2 is based on the open-source LLaMA2 model and is continually pre-trained through the proposed data-centric learning approach. Specifically, we collect and clean open-source data from the internet, developing a comprehensive data processing system and accumulating terabytes of high-quality data, which is used to train Ziya2 through three stages. The performance of Ziya2 in Chinese and English downstream tasks not only surpasses LLaMA2 but also outperforms contemporaneous open-source LLMs of similar size, which demonstrates the effectiveness of the data-centric learning approach for LLM

pre-training. In the future, we plan to continue training Ziya2, explore larger models with 70B parameters, and align Ziya2 to achieve better instructional compliance with the Ziya2-Chat model. We aim to release specialized LLMs for various domains such as writing, coding, and multimodality.

References

- Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. PaLM 2 Technical Report. *arXiv preprint arXiv:2305.10403*.
- Sid Black, Stella Biderman, Eric Hallahan, Quentin Anthony, Leo Gao, Laurence Golding, Horace He, Connor Leahy, Kyle McDonell, Jason Phang, Michael Pieler, USVSN Sai Prashanth, Shivanshu Purohit, Laria Reynolds, Jonathan Tow, Ben Wang, and Samuel Weinbach. 2022. GPT-NeoX-20B: An Open-Source Autoregressive Language Model. *arXiv preprint arXiv:2204.06745*.
- Burton H Bloom. 1970. Space/time Trade-offs in Hash Coding with Allowable Errors. *Communications of the ACM*, 13(7):422–426.
- Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language Models are Few-shot Learners. *Advances in neural information processing systems*, 33:1877–1901.
- Moses S Charikar. 2002. Similarity Estimation Techniques from Rounding Algorithms. In *Proceedings of the thirty-fourth annual ACM symposium on Theory of computing*, pages 380–388.
- Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. 2021. Evaluating Large Language Models Trained on Code. *arXiv preprint arXiv:2107.03374*.
- Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E Gonzalez, et al. 2023. Vicuna: An Open-source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality. <https://vicuna.lmsys.org> (accessed 14 April 2023).
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training Verifiers to Solve Math Word Problems. *arXiv preprint arXiv:2110.14168*.
- Together Computer. 2023. RedPajama: An Open Dataset for Training Large Language Models. <https://github.com/togethercomputer/RedPajama-Data>.

- OpenCompass Contributors. 2023. Opencompass: A Universal Evaluation Platform for Foundation Models.
- Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. FlashAttention: Fast and Memory-efficient Exact Attention with IO-Awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT (1)*, pages 4171–4186. Association for Computational Linguistics.
- Shizhe Diao, Jiaxin Bai, Yan Song, Tong Zhang, and Yonggang Wang. 2020. ZEN: Pre-training Chinese Text Encoder Enhanced by N-gram Representations. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4729–4740.
- Zhenyi Fan, Chenghao Lu, and Jie Tian. 2023. Chinese-Vicuna: A Chinese Instruction-following LLaMA-based Model.
- Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The Pile: An 800GB Dataset of Diverse Text for Language Modeling. *arXiv preprint arXiv:2101.00027*.
- Conghui He, Zhenjiang Jin, Chao Xu, Jiantao Qiu, Bin Wang, Wei Li, Hang Yan, JiaQi Wang, and Dahua Lin. 2023. Wanjian: A Comprehensive Multimodal Dataset for Advancing English and Chinese Large Models. *arXiv preprint arXiv:2308.10755*.
- Kenneth Heafield. 2011. KenLM: Faster and Smaller Language Model Queries. In *Proceedings of the sixth workshop on statistical machine translation*, pages 187–197.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring Massive Multitask Language Understanding. *arXiv preprint arXiv:2009.03300*.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring Mathematical Problem Solving with the MATH Dataset. *arXiv preprint arXiv:2103.03874*.
- Yongfeng Huang, Yanyang Li, Yichong Xu, Lin Zhang, Ruyi Gan, Jiaxing Zhang, and Liwei Wang. 2023a. MVP-Tuning: Multi-View Knowledge Retrieval with Prompt Tuning for Commonsense Reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13417–13432, Toronto, Canada.
- Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, et al. 2023b. C-Eval: A Multi-level Multi-discipline Chinese Evaluation Suite for Foundation Models. *arXiv preprint arXiv:2305.08322*.
- Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. SpanBERT: Improving Pre-training by Representing and Predicting Spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.
- Katherine Lee, Daphne Ippolito, Andrew Nystrom, Chiyuan Zhang, Douglas Eck, Chris Callison-Burch, and Nicholas Carlini. 2021. Deduplicating Training Data Makes Language Models Better. *arXiv preprint arXiv:2107.06499*.
- Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. 2023. CMMLU: Measuring Massive Multitask Language Understanding in Chinese. *arXiv preprint arXiv:2306.09212*.
- Miaofeng Liu, Yan Song, Hongbin Zou, and Tong Zhang. 2019a. Reinforced Training Data Selection for Domain Adaptation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1957–1968, Florence, Italy.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly Optimized Bert Pretraining Approach. *arXiv preprint arXiv:1907.11692*.
- Ilya Loshchilov and Frank Hutter. 2017. Decoupled Weight Decay Regularization. *arXiv preprint arXiv:1711.05101*.
- Junyu Lu, Ping Yang, Ruyi Gan, Jing Yang, and Jiaxing Zhang. 2022. Unified BERT for Few-shot Natural Language Understanding. *arXiv preprint arXiv:2206.12094*.
- Paulius Micikevicius, Sharan Narang, Jonah Alben, Gregory Diamos, Erich Elsen, David Garcia, Boris Ginsburg, Michael Houston, Oleksii Kuchaiev, Ganesh Venkatesh, et al. 2017. Mixed Precision Training. *arXiv preprint arXiv:1710.03740*.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*.
- OpenAI. 2022. Introducing ChatGPT.
- OpenAI. 2023. GPT-4 Technical Report.
- Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training Language Models to Follow Instructions with Human Feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

- Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The Refinedweb Dataset for Falcon LLM: Outperforming Curated Corpora with Web Data, and Web Data Only. *arXiv preprint arXiv:2306.01116*.
- Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar.
- Yang Ping, JunYu Lu, Ruyi Gan, Junjie Wang, Yuxiang Zhang, Pingjian Zhang, and Jiaxing Zhang. 2023. UniEX: An Effective and Efficient Framework for Unified Information Extraction via a Span-extractive Perspective. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16424–16440, Toronto, Canada.
- Han Qin, Yuanhe Tian, and Yan Song. 2021. Relation Extraction with Word Graphs from N-grams. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2860–2868, Online and Punta Cana, Dominican Republic.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language Models are Unsupervised Multitask Learners. *OpenAI blog*, 1(8):9.
- Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-text Transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.
- Samyam Rajbhandari, Jeff Rasley, Olatunji Ruwase, and Yuxiong He. 2020. Zero: Memory Optimizations toward Training Trillion Parameter Models. In *SC20: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–16. IEEE.
- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural Machine Translation of Rare Words with Subword Units. *arXiv preprint arXiv:1508.07909*.
- Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-lm: Training Multi-billion Parameter Language Models using Model Parallelism. *arXiv preprint arXiv:1909.08053*.
- Yan Song, Chia-Jung Lee, and Fei Xia. 2017. Learning Word Representations with Regularization from Prior Knowledge. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 143–152.
- Yan Song, Shuming Shi, and Jing Li. 2018. Joint Learning Embeddings for Chinese Words and Their Components via Ladder Structured Networks. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 4375–4381.
- Yan Song, Tong Zhang, Yonggang Wang, and Kai-Fu Lee. 2021. ZEN 2.0: Continue Training and Adaption for N-gram Enhanced Text Encoders. *arXiv preprint arXiv:2105.01279*.
- Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. 2021. Roformer: Enhanced Transformer with Rotary Position Embedding. *arXiv preprint arXiv:2104.09864*.
- Yuanhe Tian, Weidong Chen, Bo Hu, Yan Song, and Fei Xia. 2023. End-to-end Aspect-based Sentiment Analysis with Combinatory Categorical Grammar. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13597–13609, Toronto, Canada.
- Yuanhe Tian, Yan Song, and Fei Xia. 2022. Improving Relation Extraction through Syntax-induced Pre-training with Dependency Masking. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.
- Kushal Tirumala, Daniel Simig, Armen Aghajanyan, and Ari S Morcos. 2023. D4: Improving Llm Pre-training Via Document De-duplication and Diversification. *arXiv preprint arXiv:2308.12284*.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. LLaMA: Open and Efficient Foundation Language Models. *arXiv preprint arXiv:2302.13971*.
- Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. LLaMA 2: Open Foundation and Fine-tuned Chat Models. *arXiv preprint arXiv:2307.09288*.
- Junjie Wang, Yuxiang Zhang, Ping Yang, and Ruyi Gan. 2022. Towards No. 1 in CLUE Semantic Matching Challenge: Pre-trained Language Model Erlangshen with Propensity-Corrected Loss. *arXiv preprint arXiv:2208.02959*.
- Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2019. CCNet: Extracting High Quality Monolingual Datasets from Web Crawl Data. *arXiv preprint arXiv:1911.00359*.
- Shaohua Wu, Xudong Zhao, Tong Yu, Rongguo Zhang, Chong Shen, Hongli Liu, Feng Li, Hong Zhu, Jiangang Luo, Liang Xu, et al. 2021. Yuan 1.0: Large-scale Pre-trained Language Model in Zero-shot and Few-shot Learning. *arXiv preprint arXiv:2110.04725*.

Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, Fan Yang, et al. 2023. Baichuan 2: Open Large-scale Language Models. *arXiv preprint arXiv:2309.10305*.

Ping Yang, Junjie Wang, Ruyi Gan, Xinyu Zhu, Lin Zhang, Ziwei Wu, Xinyu Gao, Jiaying Zhang, and Tetsuya Sakai. 2022. Zero-Shot Learners for Natural Language Understanding via a Unified Multiple Choice Perspective. *arXiv preprint arXiv:2210.08590*.

Longhui Yu, Weisen Jiang, Han Shi, Jincheng Yu, Zhengying Liu, Yu Zhang, James T Kwok, Zhenguo Li, Adrian Weller, and Weiyang Liu. 2023. Metamath: Bootstrap Your Own Mathematical Questions for Large Language Models. *arXiv preprint arXiv:2309.12284*.

Sha Yuan, Hanyu Zhao, Zhengxiao Du, Ming Ding, Xiao Liu, Yukuo Cen, Xu Zou, Zhilin Yang, and Jie Tang. 2021. Wudaocorpora: A Super Large-scale Chinese Corpora for Pre-training Language Models. *AI Open*, 2:65–68.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022. GLM-130B: An Open Bilingual Pre-trained Model. *arXiv preprint arXiv:2210.02414*.

Biao Zhang and Rico Sennrich. 2019. Root Mean Square Layer Normalization. *Advances in Neural Information Processing Systems*, 32.

Jiaying Zhang, Ruyi Gan, Junjie Wang, Yuxiang Zhang, Lin Zhang, Ping Yang, Xinyu Gao, Ziwei Wu, Xiaoqun Dong, Junqing He, et al. 2022. Fengshenbang 1.0: Being the Foundation of Chinese Cognitive Intelligence. *arXiv preprint arXiv:2209.02970*.

A Pretraining Data

The following charts presents examples of unsupervised and supervised training data.

English Unsupervised Data

Text

ABC News' Good Morning America outstripped NBC News' Today by 761,000 viewers and 279,000 news demo viewers the week of April 7. It's GMA's seventh consecutive week on top of the morning infotainment show race in both metrics, and its largest demo margin in three months. GMA has ranked No. 1 in overall audience for 89 of the past 93 weeks, and No. 1 in the news demo for 25 of this season's 29 weeks to date. Today meanwhile, boasted it finished first with the younger, 18-49 year old age bracket, for the 42nd consecutive week. Today is on top of the ratings in the day part with men 25-54 this season, NBC noted — as well as adults, men and women 18-49. Today has posted seven consecutive months of ratings growth in total viewers, and both the 25-54 and 18-49 demos which NBC says is the show's biggest ratings uptick since '97. For the week, GMA clocked 5.617 million viewers — 2.212 million in the demo. Today logged 4.856 million viewers — 1.933 million in the demo. GMA bested CBS This Morning's 3.041 million viewers — 956,000 in the news demo.

Chinese Unsupervised Data

Text

你走了，带着我们还来不及开始的爱。你说我不够勇敢与果断、说我只是把喜欢寄放在你那边却从来没有想细心灌溉。但我要怎么能灌溉呢？你就像是一沃富饶的土地，大家总把他们的爱埋在你的心窝。圳水引来的滋润来不及洒落在我需要你的那些岁月，我向着有你在的阳光趋去，却如伊藤润二笔下的漩涡，只不成成了让人害怕的扭曲怪状。{...} 所以谢谢我曾经在你心里的位置那么前面、那么靠近你心的地方。

You're gone, with the love we haven't had time to start yet. You say I am not brave and decisive enough, and you say I just place my love on your side but never want to irrigate it carefully. But how can I irrigate it? You are like a fertile land, where everyone always buries their love in your heart. The nourishment brought by the water in the canal did not fall on the years when I needed you. I moved towards the sunshine with you, but like a vortex in the pen of Runji Ito, it only became a frightening twisted and strange shape {...} So thank you for being so in front of you and so close to your heart.

English Supervised Data

Text

Problem:

Human rights are moral principles or norms, which describe certain standards of human behaviour{...} Military aircraft can be either combat or non-combat: TurboJET is the brand name for the operations of the Hong Kong-headquartered Shun Tak-China Travel Ship Management Limited, which was established from the joint venture between Shun Tak Holdings Limited and China Travel International Investment Hong Kong Limited in July 1999. It operates hydrofoil ferry services in southern China. Question:use beechcraft starship

A.military

B.general aviation

C.service

Answer:B.general aviation

Chinese Supervised Data

Prompt

问题:{问题和内容} 回答:{回答的内容}

Question:{the question} Answer:{the answer}

Text

问题:

请问下面描述属于哪一种事件类型? 文章: 昨晚, 在号称“亚洲第一魔鬼主场”的天河体育场, 国足在占据天时地利人和的情况下, 半场就击垮对手, 狂轰6球{...} 这样的战绩和表现也在赛后引发了巨大的争议。

A.胜负

B.加息

C.融资

D.爆炸

回答: 答案: A.胜负

Question:

May I ask which type of event does the following description belong to? Article: Last night, at the Tianhe Stadium, which is known as the "Number One Devil's Stadium in Asia," the Chinese football team occupied Under the favorable conditions of timing, location, and people, he defeated his opponent in half time and roared 6 goals {...} Such achievements and performances have also sparked great controversy after the game.

A.Success or Failure

B.Raise Interest Rates

C.Financing

D.Explosion

Answer: The answer is: A.Success or Failure

Code

Prompt

``` (sql / python / c / etc.) + {Code} ```

### Text

``` SQL

ALTER TABLE kooste.tamperewfs_luontorastit **DROP CONSTRAINT** tamperewfs_luontorastit_pk;

ALTER TABLE kooste.tamperewfs_luontorastit **DROP CONSTRAINT** tunnus_rasti_unique;

ALTER TABLE kooste.tamperewfs_luontorastit **ADD COLUMN** kooste_id bigint **NOT NULL GENERATED**

ALWAYS AS IDENTITY;...{Code~}

```

## MetaMath

### Prompt

Question/Q:{the question} Answer/A:{the answer}

### Text

*Question:*

*The coffee shop sold 8 more than four times the number of lattes as it did teas. If 6 teas were sold how many lattes did the coffee shop sell?*

*Answer:*

*If the coffee shop sold 6 teas, then the number of lattes sold is 8 more than four times the number of teas. Four times the number of teas is  $4 * 6 = 24$ . So, the number of lattes sold is  $24 + 8 = 32$ . The answer is: 32*



## B Generation Cases

The following charts presents examples generated by Ziya2-13B with particular inputs.

### Chinese Generation

模型输入: 我国的三皇五帝分别指的是谁?

Model Input: Who are the "Three Emperors and Five Sovereigns" of our country?

### Ziya2-13B Generation

"三皇五帝"是指我国古代部落首领, 其中"三皇"一般认为指的是燧人、伏羲、神农; 而"五帝"一般指的是黄帝、颛顼、帝喾、尧、舜。

*The "Three Emperors and Five Sovereigns" refer to ancient tribal leaders in China, among which the "Three Emperors" are generally believed to refer to Suiren, Fuxi, and Shennong; The "Five Sovereigns" generally refer to Huang Di, Zhuanxu, Sovereigns Ku, Yao, and Shun.*

### English Generation

Model Input: Friends is a series of sitcoms produced by NBC in the United States,

### Ziya2-13B Generation

*and created by David Crane and Marta Kauffman. The series premiered on September 22, 1994, and ended on May 6, 2004, lasting ten seasons. With an ensemble cast starring Jennifer Aniston, Courteney Cox, Lisa Kudrow, Matt LeBlanc, Matthew Perry and David Schwimmer, the show revolves around six friends in their 20s and 30s who live in Manhattan, New York City. The series was produced by Bright Kids Productions, in association with Warner Bros. Television. The original executive producers were Kevin S. Bright, Marta Kauffman, and David Crane. Friends had received both critical acclaim and popular success throughout its original ten-season run on NBC; it received accolades such as winning the Emmy Award for Outstanding Comedy Series in 2002, and was ranked No. 25 on TV Guide's 50 Greatest TV Shows of All Time in 2002.*

### Code Generation

模型输入: 给定一个整数 $n$ , 计算所有小于等于 $n$ 的非负整数中数字1出现的个数。例如:

输入:  $n = 13$

输出: 6

分析如下: 输入 $n=13$ , 小于等于13非负整数里面包含1的数字为1, 10, 11, 12, 13, 所以数字1共出现了6次。下面是具体的python代码: ``python

Model Input: Given an integer  $n$ , calculate the number of occurrences of the number 1 in all non negative integers less than or equal to  $n$ . For example:

Input:  $n=13$

Output: 6

The analysis is as follows: input  $n=13$ , and the number containing 1 in a non negative integer less than or equal to 13 is 1, 10, 11, 12, 13, so the number 1 appears together Six times. The following is the specific Python code: `` Python

### Ziya2-13B Generation

```
import re
def countDigitOne(self , n: int) -> int:
 count = 0
 for i in range(1, n + 1):
 s = str(i)
 for x in s:
 if x == '1':
 count += 1
 return count
...
```