arXiv:submit/4846034 [cs.CL] 16 Apr 2023

Towards Better Instruction Following Language Models for Chinese: Investigating the Impact of Training Data and Evaluation

Yunjie Ji, Yan Gong, Yong Deng, Yiping Peng, Qiang Niu, Baochang Ma*, Xiangang Li
Beike Inc., Beijing, China
`{jiyunjie001,gongyan013,dengyong013,pengyiping001, niuqiang002,mabaochang001,lixiangang002}@ke.com`

Abstract

Recently, significant public efforts have been directed towards developing low-cost models with capabilities akin to ChatGPT, thereby fostering the growth of open-source conversational models. However, there remains a scarcity of comprehensive and in-depth evaluations of these models' performance. In this study, we examine the influence of training data factors, including quantity, quality, and linguistic distribution, on model performance. Our analysis is grounded in several publicly accessible, high-quality instruction datasets, as well as our own Chinese multi-turn conversations. We assess various models using a evaluation set of 1,000 samples, encompassing nine real-world scenarios. Our goal is to supplement manual evaluations with quantitative analyses, offering valuable insights for the continued advancement of open-source chat models. Furthermore, to enhance the performance and training/inference efficiency of models in the Chinese domain, we extend the vocabulary of LLaMA – the model with the closest open-source performance to proprietary language models like GPT-3 – and conduct secondary pre-training on 3.4B Chinese words. We make our model, data, as well as code publicly available[1].

1 Introduction

Large language models exhibit remarkable comprehension, generation, and reasoning capabilities that surpass those of their smaller counterparts(Brown et al., 2020; Zhang et al., 2022; Chowdhery et al., 2022; Workshop et al., 2022; Black et al., 2022; Hoffmann et al., 2022). By learning from high-quality human-generated data, these models align with human behavior and respond to human instructions, rendering them accessible to non-experts in Natural Language Processing for various applications(Ouyang et al., 2022; Bai et al., 2022; Ziegler et al., 2020; Stiennon et al., 2022; Ganguli et al., 2022; Nakano et al., 2022; Korbak et al., 2023). Nonetheless, the development of such expansive models has become increasingly exclusive, with data, models, and algorithms no longer publicly accessible.

To address this issue, the open-source community has undertaken(Ji et al., 2023; Taori et al., 2023; Peng et al., 2023; Chiang et al., 2023; Geng et al., 2023; Xu et al., 2023; Cui and Yang, 2023; Databricks, 2023b; Databricks, 2023a) numerous effective endeavors, discovering that satisfactory instruction-following performance can be achieved using smaller models with only tens of thousands of data points. The majority of these efforts are based on LLaMA(Touvron et al., 2023), a series of open-source models that yield results comparable to proprietary models like GPT-3 and Gopher but with fewer parameters. GPT-3.5(OpenAI, 2023a) and GPT-4(OpenAI, 2023b) have also been employed to generate high-quality aligned data. Stanford Alpaca(Taori et al., 2023) utilized the self-instruct method to generate 52,000 samples using GPT-3.5 based on 175 seed tasks, while LLaMA-GPT4(Peng et al., 2023) leveraged GPT-4 for higher-quality responses. Vicunna(Chiang et al., 2023) trained LLaMA on 70,000 real user conversations with ChatGPT, and Baize(Xu et al., 2023) enabled ChatGPT to generate multi-turn conversations. Table 1 details the training and evaluation methods for these open-source initiatives. Although these projects significantly advance the development of open-source chatbots, there remains a dearth of in-depth evaluation and com-

---

*Corresponding author

[1] https://github.com/LianjiaTech/BELLE

parison of these models. Current assessments are either insufficient in quantity or solely reliant on human evaluation. It is imperative to investigate how these high-quality datasets can enhance instruction-following models.

In this paper, we investigate the influence of various factors, such as the quantity, quality, and linguistic distribution of training data on model performance. Our evaluation dataset consists of 1,000 Chinese instruction samples spanning nine real user scenarios. It is worth noting that our evaluation set is still not comprehensive enough, and the scores of the model on our evaluation set may differ from the actual user experience. Building a diverse and high-quality evaluation set will be a long-term task to accomplish. Furthermore, we extend LLaMA's vocabulary and pre-trained it on Chinese corpora to boost its efficiency in processing Chinese data, resulting in a reduction of 60% training and inference time without compromising performance.

2 Related work

2.1 Large language models

Transformer-based language models, especially the generative large language models have greatly advanced the development of Natural Language Processing (Vaswani et al., 2017; Devlin et al., 2018; Lan et al., 2019; Yang et al., 2019; Dong et al., 2019; Clark et al., 2020; Raffel et al., 2020; Brown et al., 2020; Zhang et al., 2022; Chowdhery et al., 2022; Black et al., 2022; Hoffmann et al., 2022; Glaese et al., 2022; Srivastava et al., 2022). The GPT (Generative Pre-trained Transformer) family of models is a remarkable instance, and its ability to comprehend and adhere to human instructions has been enhanced by RLHF (Ouyang et al., 2022; Bai et al., 2022; Ziegler et al., 2020; Stiennon et al., 2022; Ganguli et al., 2022; Nakano et al., 2022; Korbak et al., 2023) in ChatGPT. As a result, ChatGPT has evolved from being a basic NLP task solver to a complete natural language assistant that can perform duties such as generating conversations and detecting errors in a piece of code.

2.2 Instruction tuning

Instruction-tuning is a new trend emerging from (Wei et al., 2021; Sanh et al., 2021; Mishra et al., 2021), which seeks to improve the performance of language models by teaching them to follow natural language. By formatting all tasks into natural language, generative language models are capable of dealing with almost all of NLP tasks. Early research focused on instruction tuning a general NLP task solver, and there is a trend towards converting more and more NLP datasets into a unified dataset then conducting multi-task training (Xu et al., 2022; Xie et al., 2022; Wang et al., 2022b; Khashabi et al., 2020; Min et al., 2021; Ye et al., 2021; Liu et al., 2019; Zhong et al., 2021; Chung et al., 2022). However these models still struggle with understanding general human instructions especially in real-world use cases. Until the emergence of training methods like RLHF (Ouyang et al., 2022; Bai et al., 2022; Ziegler et al., 2020; Stiennon et al., 2022), models truly began to understand various human instructions and produce good responses.

2.3 Public available chat models

Recently, there are many attempts toward public available models. Taori et al. (2023) trained a model on 52K instruction-following samples generated in the style of self-instruct (Wang et al., 2022a) using text-davinci-003, then they did a blind pairwise comparison between Alpaca and text-davinci-003 on 252 evaluation samples from the self-instruct evaluation set. Instead, based on the same seed tasks, Peng et al. (2023) generated 52K instruction-following samples using GPT-4. They assess their models on 252 user-oriented instructions by human based on the HHH criteria (Askell et al., 2021). Chiang et al. (2023) trained a chat model based on LLaMA with a dataset consisting of 70K user-shared conversations with ChatGPT (ShareGPT, 2023). They conducted a automatic evaluation with GPT-4 on 80 evaluation examples. Geng et al. (2023) not only used ChatGPT generated datasets, but also used open source human-written instruction-following data. Databricks (2023b) finetuned GPT-J with the dataset from Stanford Alpaca. To obtain a fully and truly open instruction-tuned LLM, Databricks (2023a) established a dataset of 15K human-generated instruction data and trained a model based on Pythia. Both of their model and dataset are licensed for research and commercial use. Xu et al. (2023) focused on training a model capable for multi-turn dialogue in a low-resource setting. Therefore, they leveraged ChatGPT to engage in a conversation with itself, simulating both

Table 1: A simple overview of public available chat models. More details could be found in 2.3.

Project	Base model	Training	Training data	Evaluation data	Evaluation method
Stanford alpaca	LLaMA	Full-parameter finetuning	52K text-davinci-003 generated instruction data	252 samples from self-instruct evaluation dataset	Human evaluation
LLaMA-GPT4	LLaMA	Full-parameter finetuning	52K GPT-4 generated instruction data	1. 252 user-oriented instructions 2. 80 vicuna test samples	1. Human evaluation 2. Automatic GPT-4 evaluation
Vicuna	LLaMA	Full-parameter finetuning	70K user-shared conversations with ChatGPT	80 vicuna test samples	Automatic GPT-4 evaluation
Koala	LLaMA	Full-parameter finetuning	1. Stanford alpaca 2. Anthropic HH 3. OpenAI webgpt 4. OpenAI summarization	1. 180 samples from self-instruct evaluation dataset 2. 180 real user queries that were posted online	Human evaluation
Dolly	GPT-J	Full-parameter finetuning	Stanford alpaca	-	Case demonsrtration
Dolly 2.0	Pythia	Full-parameter finetuning	15k human-written instruction data	-	Case demonsrtration
Baize	LLaMA	LoRA	15K ChatGPT generated multi-turn conversations	-	Case demonsrtration

user and AI responses. Meanwhile, they finetuned LLaMA with a parameter-efficient tuning approach(Hu et al., 2021).

2.4 Evaluation of LLMs

There are many evaluations of large language models, such as OPT (Zhang et al., 2022), BLOOM (Workshop et al., 2022), GLM (Zeng et al., 2023), and GPT-3 (Brown et al., 2020), in various tasks. (Liang et al., 2022) conducted a thorough evaluation of 30 large language models. (Qin et al., 2023) evaluated the performance of ChatGPT on various NLP tasks. (Ye et al., 2023) compared the capabilities of GPT and GPT-3.5 series models. (Bang et al., 2023) compared the reasoning, hallucination reduction, and interactivity abilities of ChatGPT in multiple languages and modalities. However many evaluation data consist of traditional NLP tasks, which differ from real-world human usage scenarios. (Srivastava et al., 2022) provided 204 tasks, which are believed to be beyond the capabilities of current large language models. (Suzgun et al., 2022) selected the 23 most difficult tasks from BIG-Bench, forming BIG-Bench Hard (BBH).

3 Collect Conversation data

ChatGPT has demonstrated a remarkable ability to generate instruction-following data, which enables the model to achieve strong performance on various tasks. However, its ability to engage in multi-turn conversations still falls short due to its limited contextual understanding based solely on this type of data. Thus, we further task ChatGPT to generate multi-turn conversation data, where it needs to generate dialogue between users and AI assistants across multiple turns. As ChatGPT tends to produce limited and repetitive dialogue scenarios such as weather queries and airplane ticket reservations, we prompt ChatGPT with the first round

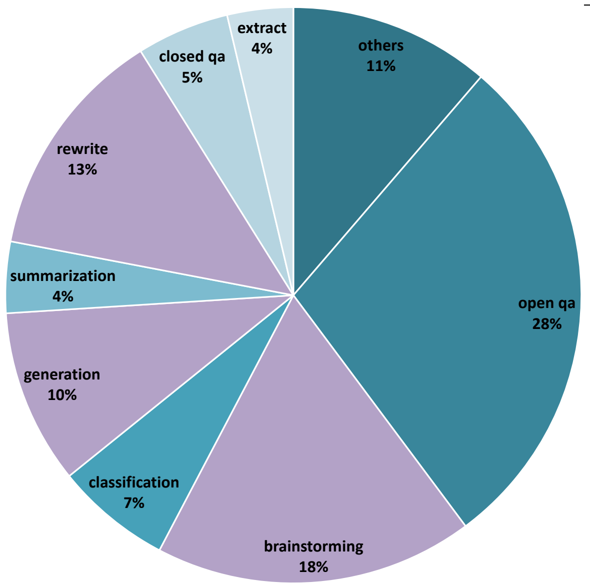


Figure 1: Task category distribution of our evaluation set. We classify math and code tasks into “others”. For rewrite, generation and brainstorming, no gold response is provided. As a consequence, ChatGPT is asked to evaluate these tasks without referring to gold responses, as we believe there are more than one reasonable responses for each instruction of these three task categories.

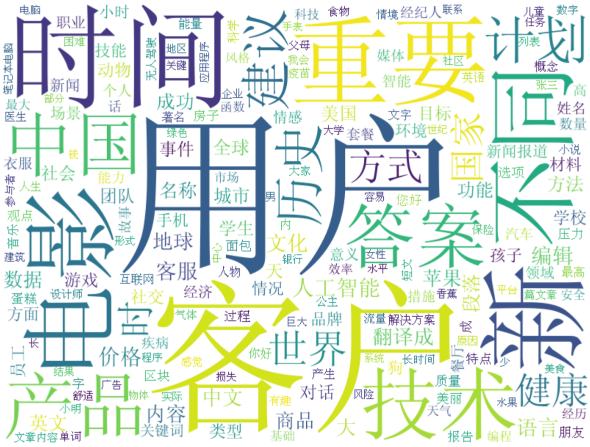


Figure 2: Word cloud of the evaluation set.

of a conversation to determine the topic of the conversation and then let ChatGPT continue the conversation accordingly.

Data Cleaning. While ChatGPT can generate high-quality data at a relatively low cost, the generated data still suffers from issues such as repetition and logical inconsistency. To improve the quality of the generated data, we first remove duplicates at both token-level and semantic-level. Next, we use metrics such as perplexity (PPL) to select high-quality data and ensure diversity in the data by examining word frequency distributions.

4 Evaluation data

The evaluation data used in this paper is refined from Ji et al. (2023). We deduplicate the original evaluation data semantically and re-classify math and code tasks into others. There are two reasons: firstly, these tasks are relatively difficult and current open-source models do not perform well on them, which could affect our evaluation of other abilities, secondly, ChatGPT is not reliable enough in evaluating these two tasks, which could lead to biased experimental results. Figure 1 depicts the task category distribution, which is not balanced, so we use macro-averaging when calculating overall scores. Figure 2 shows the word cloud of the evaluation set. We find that ChatGPT tends to generate data on certain specific topics. Figure 3 shows the length of evaluation samples.

5 Extend vocabulary

Due to the lack of optimization for Chinese language in LLaMA’s vocabulary construction, a Chinese character may be split into 2 to 3 byte tokens, which severely affects the model’s fine-tuning and inference speed on Chinese data(Cui and Yang, 2023). In order to address this issue, we train a tokenizer based on the byte-pair encoding (BPE) algorithm using sentencepiece(Kudo and Richardson, 2018) on 12M lines of Chinese text, and set its vocabulary size to 50K. We merge the trained new vocabulary with the original LLaMA vocabulary, resulting in a new vocabulary of 79,458 tokens. After that, we resize word embeddings and further pretrain LLaMA on 3.4B Chinese words with other parameters fixed. We test the extended tokenizer and the original tokenizer on 5,000 lines of Chinese text, and the average tokens of a line reduces from 733 to 291.

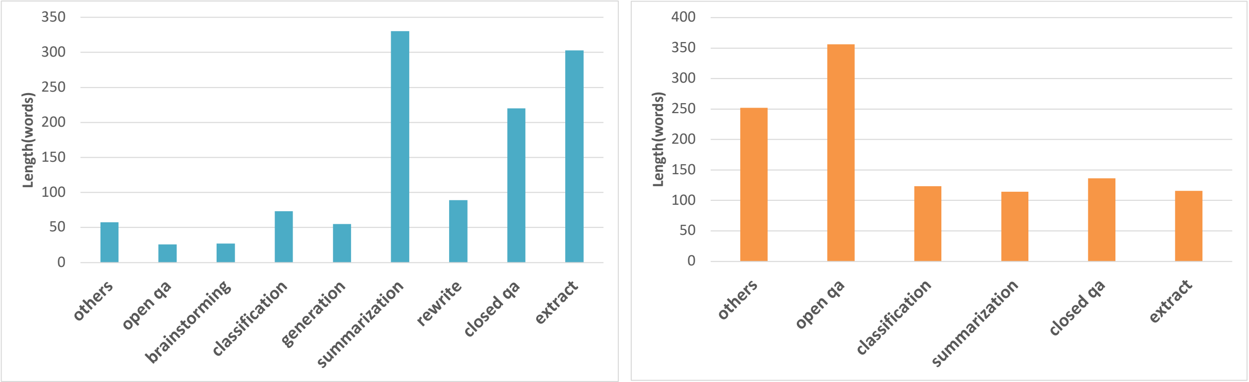
6 Experiments

6.1 Base model

In our experiments, two base models with 7B parameters are adopted:

LLaMA(Touvron et al., 2023), which is released by Meta AI.

LLaMA-EXT, which is obtained by extending the vocabulary of the vanilla LLaMA and further pre-train on 3.4B Chinese words in which only word embeddings are updated.



(a) Average instruction length

(b) Average gold response length

Figure 3: (a) shows average length of instructions, (b) show average length of gold responses.

Table 2: This table shows performances of models instruction-tuned with different base models and training data. We divide experimental results according to training data factors which may have great impact on model performance. **Score_w/0_others** is macro-averaged on categories excluding math and code. Please refer to Appendix a for scores on each category and averaged on all task categories.

Factor	**Base model**	**Training data**	**Score_w/o_others**
Extending vocabulary	LLaMA-EXT	zh(alpaca-3.5&4) + sharegpt	**0.670**
	LLaMA	zh(alpaca-3.5&4) + sharegpt	0.652
Data quality	LLaMA-EXT	zh(alpaca-3.5)	0.642
		zh(alpaca-4)	**0.693**
Data linguistic distribution	LLaMA-EXT	zh(alpaca-3.5&4)	**0.679**
		en(alpaca-3.5&4)	0.659
		zh(alpaca-3.5&4) + sharegpt	0.670
		en(alpaca-3.5&4) + sharegpt	0.668
Data quantity	LLaMA-EXT	zh(alpaca-3.5&4) + sharegpt	0.670
		zh(alpaca-3.5&4) + sharegpt + belle-3.5	**0.762**
-	ChatGPT	-	0.824

6.2 Training settings

All of the instruction-following models are fine-tuned with the same hyper-parameter settings as the open source projects with the exception of a smaller learning rate. No dev set is used while training models, and the last checkpoint is adopted for evaluation. Table 3 lists the hyper parameters. We conduct experiments on 8 A100 GPUs, each has 80G memory.

6.3 Dataset

Our experiments are conducted using six datasets, of which five are publicly available and one is proprietary.

Alpaca-3.5-en(Taori et al., 2023), which is released by Stanford Alpaca and consists of 52K instruction-following samples. These samples are generated by text-davinci-003.

Table 3: Hyper-parameter settintgs, which are the same as previous open-released chat model with the exception of a smaller learning rate.

Hyper parameter	**Value**
Precision	bf16
Epochs	3
Batch size	32
Learning rate	5e-6
Weight decay	0
Warmup ratio	0.03
LR scheduler type	cosine
Max length	2048

Alpaca-3.5-zh(Cui and Yang, 2023), which is the translated Chinese version of alpaca-3.5-en.

Alpaca-4-en, Alpaca-4-zh(Peng et al., 2023), which are released by LLaMA-GPT4, both containing 52K instruction -following samples. These samples are generated by GPT-4. To obtain alpaca-4-zh, Peng et al. (2023) first used ChatGPT to translate 52K instructions into Chinese then asked GPT-4 to answer them in Chinese.

ShareGPT(ShareGPT, 2023), which are user-shared conversations with ChatGPT, consisting of 8.3K samples. We conduct three steps of data cleaning (Chiang et al., 2023). Only English and Chinese conversations are kept. Besides, conversations are divided into smaller segments with a maximum length of 2048 tokens. Finally, we derive 120,009 conversations.

Belle-3.5, which is our own dataset, consisting of instruction-following samples and multi-turn conversations. This dataset contains 500,000 samples that are filtered out from 2.3M raw data with the cleaning method mentioned in section 3.

To simplify dataset names while conducting experiments under different data settings, we define two functions for identifying the language of the given dataset. `zh(d)` means the Chinese version of `d` and `en(d)` means the English one.

6.4 Metric

ChatGPT is asked to evaluate responses generated by instruction-following models. For all instructions, ChatGPT gives a score between 0 and 1, where score 0 is the worst and score 1 is the best. In order to reduce randomness, we set the temperature to 0.001 for model generation. Evaluation is achieved by invoking gpt-3.5-turbo API at the time of April 15, 2023.

We calculate model's scores for each task category and derive its overall performance on the evaluation set using macro average across these categories. Given ChatGPT's limitations in evaluating mathematical and coding tasks, we separately compute the scores that include all categories (denoted as **score**) and those that exclude these two categories (denoted as **score_w/o_others**). Table 2 presents the main experimental results. Please refer to Appendix 8.1 for the detailed scores on each task category.

6.5 Main results

Extending vocabulary With the same training data, finetuning on LLaMA-EXT outperforms finetuning on LLaMA, which could be due to the continued pre-training on 3.4B Chinese words, thus augmenting LLaMA's understanding of the Chinese language.

Data quality The performance of LLaMA-EXT, when trained with alpaca-4-zh, surpasses that of the model trained with alpaca-3.5-zh, thus highlighting the critical impact of high-quality training data on enhancing model performance.

Linguistic distribution Comparing the performance of LLaMA-EXT trained on zh(alpaca-3.5&4) with the performance of LLaMA-EXT trained on en(alpaca-3.5&4), it is observed that utilizing Chinese instruction-following data results in superior performance within Chinese testing scenarios. Interestingly, the disparity in performance between these two models is not substantial, which suggests that the model's capacity to comprehend instructions in one language can be effectively transferred to another language, owing to its inherent multilingual capabilities rooted from the pre-trained language model. The scores of zh(alpaca-3.5&4)+sharegpt and en(alpaca-3.5&4)+sharegpt are nearly identical. This signifies that the addition of a modest quantity of Chinese data (which comes from sharegpt) to the English training data can dramatically enhance performance within the Chinese context. However, it is also discovered that incorporating sharegpt into zh(alpaca-3.5&4) does not yield any further improvements. This may be attributed to our inability to validate the model's proficiency in multi-turn conversations using our evaluation set.

Data quantity In terms of training data quantity, an increase in data quantity has been shown to significantly improve performance. It is worth noting that such huge improvement may partly come from the similar distribution between belle-3.5 and our evaluation data. The categories, topics and complexity of evaluation data will brings great impact on evaluation results.

Compare with ChatGPT Lastly, we identify a considerable performance gap between our model and ChatGPT. Considering the limited evaluation capabilities of ChatGPT and the incompleteness of our evaluation data, it is anticipated that these disparities might be even larger than the score differences observed within the evaluation set. This serves as motivation for us to persist in improving our model.

6.6 Challenge of building evaluation set

Through case analysis, we discover that our evaluation set faces limitations in its comprehensiveness, leading to an incongruity between the improvements of model scores and actual user experience.

Constructing a high-quality evaluation set presents a significant challenge, as it necessitates the inclusion of a diverse range of usage scenarios while maintaining a balanced difficulty gradient. If the evaluation examples predominantly consist of difficult cases, the performance of all models will be poor, making it challenging to discern the effects of various training strategies. Conversely, if the evaluation examples are predominantly easy, the assessment will lose their comparative value. Moreover, it is imperative to ascertain that the evaluation data remains distinct from the training data.

Based on these observations, we caution against the assumption that the model has achieved a performance level comparable to ChatGPT solely by obtaining favorable results on a limited number of test samples. We believe that prioritizing the continuous development of a comprehensive evaluation set is of great importance.

6.7 Case demonstration

In Table 4, we list several evaluation examples and corresponding model responses to qualitatively demonstrate the gap between ChatGPT and our instruction-following model. Our model is obtained by finetuning LLaMA-EXT with zh(alpaca-3.5&4)+sharegpt+belle-3.5. In our study, we find that our model tends to generate longer responses, which may be due to the presence of GPT-4 generated or multi-turn dialogue data in our training set. In the first example, our model made an error by classifying birds as mammals, while ChatGPT classified fish as reptiles, which is also not entirely accurate. For the following two math problems, although our model correctly identified the steps, it made mistakes in numerical calculations. As for the final example, both models generated reasonable responses, but our model showed slight content repetition. This may be related to our setting of a low temperature (0.001) during generation. When we increase the temperature to 0.5, we observe a significant increase in diversity of responses for the same task.

7 Conclusion

In conclusion, this study addresses the growing need for comprehensive evaluations of open-source conversational models by investigating the influence of various training data factors, such as quantity, quality, and linguistic distribution. By utilizing publicly accessible high-quality instruction datasets and Chinese multi-turn conversations, we assess different models on a evaluation set of 1,000 samples across nine real-world scenarios. We also conclude several challenges of building a comprehensive evaluation dataset and argue the necessity of prioritizing the development of such evaluation set. Moreover, this study extends the vocabulary of LLaMA and conducts secondary pre-training with 3.4B Chinese words to enhance its performance and efficiency in the Chinese domain. This results in a 60% reduction in training and inference time without sacrificing performance. By making the model, data, and code publicly available, this research contributes to the ongoing efforts of the open-source community to develop more accessible and efficient conversational models especially for Chinese.

Table 4: Examples of responses generated by ChatGPT and our model.

Instruction	**Our model**	**ChatGPT**
将下列五个词分为两个组别，每个组别都有一个共同点：狗、猫、鸟、鱼、蛇。	将这五个词分为两个组别，可以基于它们的共性来进行分类。以下是一种可能的分类方式： 组别一：哺乳动物 - 狗 - 猫 - 鸟 这三个词都是哺乳动物，它们都有毛发和乳腺，能够哺育幼崽。 组别二：爬行动物 - 鱼 - 蛇 这两个词都是爬行动物，它们都有鳞片和冷血，并且通常通过产卵来繁殖后代。 这种分类方式基于这些动物的共性，即它们都是脊椎动物，并且都有特定的生理特征和行为习惯。这种分类方式也有助于我们更好地理解这些动物，以及它们与其他动物之间的相似性和差异性。	组别1：狗、猫、鸟共同点：都是宠物 组别2：鱼、蛇共同点：都属于爬行动物，没有四肢。
请给出以下数学问题的答案：$10 \div 2 \times (5-2)^2 + 7$	首先，按照数学运算的优先级，需要先计算括号里的表达式，即$5-2=3$，然后将3的平方计算，即$3^2=9$。接下来，需要先计算乘法，即$10 \div 2 \times 9 = 45$。最后，需要先计算除法，即$10 \div 2 = 5$，然后将乘法和除法的结果相加，即$45+5=54$。因此，$10 \div 2 \times (5-2)^2 + 7$的答案是54。	答案是52。计算步骤如下：1. 先计算括号内的表达式$5-2=3$。2. 再计算括号外的除法$10 \div 2 = 5$。3. 接着计算括号外的乘方$3^2=9$。4. 最后加上常数项得到结果$5 \times 9 + 7 = 52$。
学校买来6箱乒乓球拍，每箱8副，每副9元．一共花了多少钱?	学校买了6箱乒乓球拍，每箱8副，每副9元。为了计算总共花费的钱数，我们需要将每箱的8副乒乓球拍乘以每副的9元，然后将结果相加。6箱×8副×9元/副= 54元因此，学校一共花了54元购买这些乒乓球拍。	一共买来（6×8）=48副乒乓球拍，花费的钱数为：48×9=432元。答案：432元。
根据以下资料，为公司编写一个招聘广告：公司名称：ABC科技，职位：软件工程师，要求：熟练掌握Java，有2年以上工作经验。	ABC科技招聘软件工程师！ 我们正在寻找一位熟练掌握Java，有2年以上工作经验的人才加入我们的团队。作为一家领先的技术公司，我们致力于为客户提供最优质的软件解决方案。 如果您热爱编程，对Java有着深入的理解，并且拥有2年以上相关工作经验，我们非常欢迎您加入我们的团队。 作为软件工程师，您将负责开发和维护我们的软件产品，与团队成员紧密合作，确保项目按时交付，并为客户提供最优质的技术支持。 如果您符合以上条件，请尽快联系我们！我们期待您的加入！	ABC科技是一家专注于软件开发的公司，现招聘软件工程师一名。我们要求申请者熟练掌握Java编程语言，具备至少2年以上的相关工作经验。如果你有扎实的计算机技能、良好的团队合作能力和独立解决问题的能力，欢迎加入我们！我们将向你提供有竞争力的薪酬待遇、良好的职业发展空间和良好的工作环境。如果你对此职位感兴趣，请将简历发送至我们的邮箱，并注明“软件工程师申请”。我们期待您的加入！

References

[Askell et al.2021] Amanda Askell, Yuntao Bai, Anna Chen, et al. 2021. A general language assistant as a laboratory for alignment, December.

[Bai et al.2022] Yuntao Bai, Saurav Kadavath, Sandipan Kundu, et al. 2022. Constitutional ai: Harmlessness from ai feedback, December.

[Bang et al.2023] Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.

[Black et al.2022] Sid Black, Stella Biderman, Eric Hallahan, et al. 2022. Gpt-neox-20b: An open-source autoregressive language model, April.

[Brown et al.2020] Tom B. Brown, Benjamin Mann, Nick Ryder, et al. 2020. Language models are few-shot learners, July.

[Chiang et al.2023] Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90quality, March.

[Chowdhery et al.2022] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, et al. 2022. Palm: Scaling language modeling with pathways, October.

[Chung et al.2022] Hyung Won Chung, Le Hou, Shayne Longpre, et al. 2022. Scaling instruction-finetuned language models, October.

[Clark et al.2020] Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

[Cui and Yang2023] Yiming Cui and Ziqing Yang. 2023. Chinese llama and alpaca llms. `https://github.com/ymcui/Chinese-LLaMA-Alpaca`.

[Databricks2023a] Databricks. 2023a. Free dolly: Introducing the world's first truly open instruction-tuned llm. Blog post, April.

[Databricks2023b] Databricks. 2023b. Hello dolly: Democratizing the magic of chatgpt with open models. Blog post, March.

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

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

[Ganguli et al.2022] Deep Ganguli, Liane Lovitt, Jackson Kernion, et al. 2022. Red teaming language models to reduce harms: Methods, scaling behaviors, and lessons learned.

[Geng et al.2023] Xinyang Geng, Arnav Gudibande, Hao Liu, Eric Wallace, Pieter Abbeel, Sergey Levine, and Dawn Song. 2023. Koala: A dialogue model for academic research. Blog post, April.

[Glaese et al.2022] Amelia Glaese, Nat McAleese, Maja Trebacz, John Aslanides, Vlad Firoiu, Timo Ewalds, Maribeth Rauh, Laura Weidinger, Martin Chadwick, Phoebe Thacker, et al. 2022. Improving alignment of dialogue agents via targeted human judgements. *arXiv preprint arXiv:2209.14375*.

[Hoffmann et al.2022] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, et al. 2022. Training compute-optimal large language models, March.

[Hu et al.2021] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

[Ji et al.2023] Yunjie Ji, Yong Deng, Yan Gong, Yiping Peng, Qiang Niu, Lei Zhang, Baochang Ma, and Xiangang Li. 2023. Exploring the impact of instruction data scaling on large language models: An empirical study on real-world use cases. *arXiv preprint arXiv:2303.14742*.

[Khashabi et al.2020] Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. Unifiedqa: Crossing format boundaries with a single qa system. *arXiv preprint arXiv:2005.00700*.

[Korbak et al.2023] Tomasz Korbak, Kejian Shi, Angelica Chen, et al. 2023. Pretraining language models with human preferences, February.

[Kudo and Richardson2018] Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

[Lan et al.2019] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

[Liang et al.2022] Percy Liang, Rishi Bommasani, Tony Lee, et al. 2022. Holistic evaluation of language models, November.

[Liu et al.2019] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*.

[Min et al.2021] Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2021. Metaicl: Learning to learn in context. *arXiv preprint arXiv:2110.15943*.

[Mishra et al.2021] Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2021. Cross-task generalization via natural language crowdsourcing instructions. *arXiv preprint arXiv:2104.08773*.

[Nakano et al.2022] Reiichiro Nakano, Jacob Hilton, Suchir Balaji, et al. 2022. Webgpt: Browser-assisted question-answering with human feedback, June.

[OpenAI2023a] OpenAI. 2023a. Chatgpt: Optimizing language models for dialogue. Blog post.

[OpenAI2023b] OpenAI. 2023b. Gpt-4 technical report. Blog post.

[Ouyang et al.2022] Long Ouyang, Jeff Wu, Xu Jiang, et al. 2022. Training language models to follow instructions with human feedback, March.

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

[Qin et al.2023] Chengwei Qin, Aston Zhang, Zhuosheng Zhang, et al. 2023. Is chatgpt a general-purpose natural language processing task solver?, February.

[Raffel et al.2020] 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.

[Sanh et al.2021] Victor Sanh, Albert Webson, Colin Raffel, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv:2110.08207 [cs]*, October.

[ShareGPT2023] ShareGPT. 2023. Share your wildest chatgpt conversations with one click. Blog post.

[Srivastava et al.2022] Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*.

[Stiennon et al.2022] Nisan Stiennon, Long Ouyang, Jeff Wu, et al. 2022. Learning to summarize from human feedback, February.

[Suzgun et al.2022] Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. 2022. Challenging big-bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*.

[Taori et al.2023] Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

[Touvron et al.2023] Hugo Touvron, Thibaut Lavril, Gautier Izacard, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

[Vaswani et al.2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

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

[Wang et al.2022b] Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, et al. 2022b. Super-naturalinstructions: Generalization via declarative instructions on 1600+ nlp tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109.

[Wei et al.2021] Jason Wei, Maarten Bosma, Vincent Y. Zhao, et al. 2021. Finetuned language models are zero-shot learners. *arXiv:2109.01652 [cs]*, September.

[Workshop et al.2022] BigScience Workshop, Teven Le Scao, Angela Fan, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model, December.

[Xie et al.2022] Tianbao Xie, Chen Henry Wu, Peng Shi, Ruiqi Zhong, Torsten Scholak, Michihiro Yasunaga, Chien-Sheng Wu, Ming Zhong, Pengcheng Yin, Sida I Wang, et al. 2022. Unifiedskg: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. *arXiv preprint arXiv:2201.05966*.

[Xu et al.2022] Hanwei Xu, Yujun Chen, Yulun Du, Nan Shao, Yanggang Wang, Haiyu Li, and Zhilin Yang. 2022. Zeroprompt: Scaling prompt-based pretraining to 1,000 tasks improves zero-shot generalization. *arXiv preprint arXiv:2201.06910*.

[Xu et al.2023] Canwen Xu, Daya Guo, Nan Duan, and Julian McAuley. 2023. Baize: An open-source chat model with parameter-efficient tuning on self-chat data. *arXiv preprint arXiv:2304.01196*.

[Yang et al.2019] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

[Ye et al.2021] Qinyuan Ye, Bill Yuchen Lin, and Xiang Ren. 2021. Crossfit: A few-shot learning challenge for cross-task generalization in nlp. *arXiv preprint arXiv:2104.08835*.

[Ye et al.2023] Junjie Ye, Xuanting Chen, Nuo Xu, Can Zu, Zekai Shao, Shichun Liu, Yuhan Cui, Zeyang Zhou, Chao Gong, Yang Shen, et al. 2023. A comprehensive capability analysis of gpt-3 and gpt-3.5 series models. *arXiv preprint arXiv:2303.10420*.

[Zeng et al.2023] Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, Weng Lam Tam, Zixuan Ma, Yufei Xue, Jidong Zhai, Wenguang Chen, Zhiyuan Liu, Peng Zhang, Yuxiao Dong, and Jie Tang. 2023. GLM-130b: An open bilingual pre-trained model. In *The Eleventh International Conference on Learning Representations (ICLR)*.

[Zhang et al.2022] Susan Zhang, Stephen Roller, Naman Goyal, et al. 2022. Opt: Open pre-trained transformer language models, June.

[Zhong et al.2021] Ruiqi Zhong, Kristy Lee, Zheng Zhang, and Dan Klein. 2021. Adapting language models for zero-shot learning by meta-tuning on dataset and prompt collections. *arXiv preprint arXiv:2104.04670*.

[Ziegler et al.2020] Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, et al. 2020. Fine-tuning language models from human preferences, January.

8 Appendix A

8.1 Detailed evaluation scores

Table 5: Detailed scores on each task category, which is complementary to 2.

Factor	**Base model**	**Training data**	**others**	**rewrite**	**classif-ication**	**generation**	**summari-zation**	**extract**	**open qa**	**brain-storming**	**closed qa**	**macro ave**	**macro ave_ w/o_others**
Extending vocabulary	LLaMA -EXT	zh(alpaca-3.5&4) + sharegpt	0.419	0.858	0.655	0.897	0.663	0.456	0.422	0.837	0.577	0.643	0.670
	LLaMA	zh(alpaca-3.5&4) + sharegpt	0.442	0.828	0.655	0.853	0.743	0.444	0.355	0.803	0.537	0.629	0.652
Data quality	LLaMA -EXT	zh(alpaca-3.5)	0.338	0.789	0.691	0.825	0.624	0.423	0.358	0.805	0.616	0.608	0.642
		zh(alpaca-4)	0.419	0.825	0.626	0.918	0.690	0.559	0.451	0.879	0.593	0.662	0.693
Data linguistic distribution	LLaMA -EXT	zh(alpaca-3.5&4)	0.412	0.807	0.637	0.889	0.709	0.489	0.445	0.814	0.645	0.650	0.679
		en(alpaca-3.5&4)	0.376	0.760	0.606	0.894	0.744	0.489	0.376	0.899	0.502	0.627	0.659
		zh(alpaca-3.5&4) + sharegpt	0.419	0.858	0.655	0.897	0.663	0.456	0.422	0.837	0.577	0.643	0.670
		en(alpaca-3.5&4) + sharegpt	0.508	0.774	0.632	0.818	0.691	0.578	0.380	0.819	0.653	0.650	0.668
Data quantity	LLaMA -EXT	zh(alpaca-3.5&4) + sharegpt	0.419	0.858	0.655	0.897	0.663	0.456	0.422	0.837	0.577	0.643	0.670
		zh(alpaca-3.5&4) + sharegpt + belle-3.5	0.566	0.904	0.820	0.984	0.753	0.461	0.564	0.938	0.672	0.740	0.762
-	ChatGPT	-	0.875	0.861	0.813	0.971	0.795	0.767	0.690	0.944	0.751	0.830	0.824