ChineseAlpacaEval: A Benchmark for Chinese Instruction-following Large Language Models

Abstract

Currently, there is a lack of evaluations on whether Large language models (LLMs) can align with the preferences of Chinese users. Similar evaluations are mostly for English, while Chinese LLM evaluations focus more on evaluating the knowledge level of Chinese. Therefore, we develop ChineseAlpacaEval to fill this gap, an automated Chinese LLM evaluation benchmark that is based on AlpacaEval. The benchmark data and code for automatic evaluation are publicly available at https://github.com/CrossmodalGroup/ChineseAlpacaEval.

1 Introduction

Benefiting from the further advancement in modeling language distributions, Large language models (LLMs) not only excel in addressing a wide range of traditional Natural Language Processing (NLP) tasks but also demonstrate potential for generating responses to natural language instructions that align with human preferences (Zhao et al., 2023). However, LLMs with higher scores on traditional NLP benchmarks such as MMLU (Hendrycks et al., 2020) do not necessarily have higher user preferences. This is because these benchmarks have a limited form of task and can only measure part of LLM's capabilities. Consequently, researchers have introduced various automated evaluation benchmarks such as MT-Bench (Zheng et al., 2023) and AlpacaEval (Li et al., 2023b) to evaluate how well LLM is aligned with user preferences. Nonetheless, these benchmarks focus primarily on English and lack comprehensive tests for other languages like Chinese.

Currently, several benchmarks for Chinese LLMs have emerged, including Ceval (Huang et al., 2023), CMMLU (Li et al., 2023a), GAOKAO-Bench(Zhang et al., 2023), etc. However, most of these benchmarks evaluate Chinese knowledge of models through multiple-choice formats or by integrating various traditional Chinese NLP tasks. There is a lack of evaluation of whether LLMs can follow instructions from Chinese users and generate responses that match Chinese user preferences.

Therefore, we propose the ChineseAlpacaEval benchmark, which is based on the widely recognized AlpacaEval. It can be used to evaluate the ability of LLMs to generate responses that match Chinese users' preferences. We use large language models such as GPT-4 (OpenAI, 2023) for automated evaluations. We convert the AlpacaEval test instruction set into Chinese in three steps: translation, Chinese background replacement, and human correction. This process ensures the instructions fit better with the Chinese knowledge background while maintaining a high level of fluency. We compare the responses of the target model with those of the text-davinci-003[1] on the ChineseAlpacaEval instruction set and calculate the win rate as the benchmark score.

2 Data Construction

The source of our data is the AlpacaEval open-source 805 instructions, which is a simplification of the AlpacaFarm (Dubois et al., 2023) evaluation set. To build a Chinese evaluation instruction set using the AlpacaEval dataset, we performed three steps: translation, Chinese context replacement, and manual correction. As a result, we obtained 795 Chinese instructions and used some LLMs to generate their reference responses.

2.1 Translation

In this step, we utilize GPT-4 to translate instructions in English into Chinese. To avoid a noticeable foreign accent within the translation, we design relevant prompts that emphasize the importance of ensuring smooth and natural localization aligned with the speech habits of Chinese users. We use GPT-4 for this process. Figure 1 displays the prompt

[1] https://platform.openai.com/docs/models/gpt-3-5

[System]:
You are a helpful assistant. Your task is to refer to the queries provided by users in other languages and create a query that a Chinese user would ask. You are required to make it fluent, while also localizing them as much as possible to fit the speech habits of mainland China. However, proper nouns cannot be changed, and foreign names, place names, code snippets, abbreviations, etc. do not need to be converted to Chinese.

[Instruction]:
Please refer to the query provided by users in other languages and create a query that a Chinese user would ask. Note that proper nouns cannot be changed. You only need to give the converted text, not the explanatory information, hints or prompts. Here is the query to be converted: "{query}"

Please write below how a Chinese user would express this query:

Figure 1: The prompt for the translation step of data construction.

implemented to translate. The '[System]' part signifies system content, the '[Instruction]' part represents the input prompt, 'query' indicates the English instruction input, and '[Response]' shows the output from GPT-4.

[System]:
You are a helpful assistant. Your task is to refer to the queries provided by users in other languages and create a query that a Chinese user would ask. You are required to make it fluent, while also localizing them as much as possible to fit the speech habits of mainland China.

[Instruction]:
请参考其它国家用户提出的问题，创建一个中国用户会提出的问题。尽量将问题对象替换为中国的，并使其符合中国背景。注意您需要使问题内容更多样化，同时也需要符合事实。没有指定特定国家背景或是背景为著名人物、事物、地点的指令则不需要进行中文本地化。您只需要给出转换后的问题，不需要任何提示信息。
这里是待转换的问题: "{query}"

请在下面写出转换后的问题:

Figure 2: The prompt for the Chinese background replacement step of data construction.

2.2 Chinese Background Replacement

After translating the text, it has become apparent that the instructions lack adequate localization and background knowledge of the Chinese language. Consequently, directly utilizing the translated instructions as an evaluation instruction set for Chinese LLMs would result in an inadequate evaluation of Chinese users' preferences. As such, we intend to use GPT-4 to substitute some locations, individuals, or inclusions in the instructions that were originally from other countries with Chinese equivalents. We utilize the prompt displayed in Figure 2 for the Chinese localization of instructions. Our preliminary tests demonstrated that adopting the English prompt for Chinese background replacement was inadequate, so we replaced the input prompt with Chinese to keep the input and output languages consistent.

2.3 Human Calibration

In order to further improve the quality of the Chinese instructions, we manually correct the instructions after the Chinese background replacement in the second step. The human correction mainly focuses on four aspects:

- Coherence: Some translated instructions lacked coherence and required manual adjustments to the word order.
- Diversity: Certain instructions had repetitive background knowledge. We manually rephrased these instructions to introduce more diversity.
- Illusions: Some instructions made statements that were inconsistent with facts after Chinese background replacement. These statements need to be manually corrected.
- Excessive localization: Some of the instructions that do not specify a particular country context or have a famous person, thing, or place in the context do not have a need for Chinese localization. In such cases, those instructions were replaced with the results directly translated in the first step.

During the human calibration process, we found that some instructions did not make sense on their own. Therefore, we excluded a portion of the instructions. As a result, the final Chinese instruction set contained a total of 795 instructions.

3 Evaluation

Our evaluation methodology is based on AlpacaEval, which enables the automatic evaluator to choose its preferred output between two model outputs. Currently, we employ GPT-4 as our automatic evaluator, using the Chinese prompt shown in Figure 3 for evaluation. This prompt receives

```
我希望你创建一个针对大语言模型中文能力的排行榜。为此，我将提供给你中文指令以及两个模型对应的中文回复。请根据中文用户的偏好对这些模型进行排名。所有的输入和输出都应该是Python字典。

这里是提供给模型的指令：
{{
    "instruction": """{instruction}""",
}}

这里是两个模型的输出：
[
    {{
        "model": "model_1",
        "answer": """{output_1}"""
    }},
    {{
        "model": "model_2",
        "answer": """{output_2}"""
    }}
]

现在请你按照中文答案的质量对模型进行排序，以便排在第 1 位的模型输出结果最好。然后返回模型名称和排名的列表，即生成以下输出：
[
    {{'model': <model-name>, 'rank': <model-rank>}},
    {{'model': <model-name>, 'rank': <model-rank>}}
]

你的回答必须是一个有效的Python字典列表，并且除此之外不应包含任何其他内容，因为我们将直接在Python中执行它。请提供大多数中文用户会给出的结果。
```

Figure 3: The prompt for the evaluator to compare the Chinese answer quality of two LLMs. The input to the evaluator is an instruction and the corresponding responses for the two LLMs, and the output is a list of Python dictionaries containing the rankings of the LLMs.

an instruction and a pair of model outputs, corresponding to the two models being evaluated.

Before inputting the outputs into the evaluator, we randomize the order of the model responses to avoid positional bias. The evaluator references the question included in the instruction and ranks the two models based on the quality of their Chinese replies, ultimately providing the name of the model with the highest output quality. In order to enhance the consistency of evaluations, we set the temperature to 0 of GPT-4, which serves as our evaluator.

3.1 Metric

We use the Win rate as a measure of the model's Chinese conversation capability. To calculate the win rate, we collect the responses from text-davinci-003 and the model to be evaluated for each instruction in the Chinese instruction set, then let the automatic evaluator judge which one has better Chinese response quality. We count the number of times the automatic evaluator prefers the target model and compute the ratio of the output of the target model that is better than the output of text-davinci-003, which represents the model's win rate.

We observed that when evaluating according to the prompt in Figure 3, there is a minor possibility that GPT-4 may produce an incorrect format resulting in parsing failure. We refer to this situation as "Error." Consequently, in the final results, there is a small portion where the sum of the win rate and the loss rate of certain models is less than 100%. In the final ranking, if the win rates are equal, models with lower loss rates take precedence. The win rate for text-davinci-003 is set at 50%.

3.2 Leaderboard

In total, we evaluated more than 20 models, they can be classified as (1) commercial closed-source models, accessed using APIs, including GPT-4, GPT-3.5-turbo[2], text-davinci-003, ChatGLM_pro (Zeng et al., 2022), Ernie_bot[3], Spark[4] and Qwen_turbo[5]. (2) open-source models, accessed using model weights, including models other than (1). Among these models, some of them were specifically trained using Chinese data, such as ChatGLM (Du et al., 2022), Ernie_bot,

[2]https://platform.openai.com/docs/models/gpt-3-5
[3]https://yiyan.baidu.com/
[4]https://xinghuo.xfyun.cn/
[5]https://qianwen.aliyun.com/

Model Name	Win Rate(%)	Lose Rate(%)	Error(%)	Length	Rank
GPT-4-0613	91.19	8.81	0.0	308	1
Chatglm_pro	90.57	9.43	0.0	430	2
GPT-3.5-turbo-0613	89.43	10.57	0.0	307	3
Ernie_bot	88.81	11.19	0.0	454	4
Baichuan-13B-chat	85.28	14.72	0.0	449	5
Baichuan2-13B-chat	85.16	14.72	0.13	450	6
Spark	82.89	16.98	0.13	327	7
Xwin-LM-13B-V0.1	82.14	17.86	0.0	466	8
Chinese-alpaca-2-13b	79.87	20.13	0.0	491	9
Qwen-14B-Chat	76.23	23.52	0.25	295	10
WizardLM-13B-V1.2	75.6	24.4	0.0	411	11
ChatGLM2-6B	74.34	25.66	0.0	421	12
BELLE-Llama2-13B-chat	70.82	29.18	0.0	293	13
Qwen-7B-Chat	69.69	30.31	0.0	275	14
ChatGLM-6B	68.18	31.82	0.0	401	15
Qwen_turbo	65.66	34.34	0.0	248	16
Firefly-llama2-13b-chat	52.2	47.8	0.0	261	17
text-davinci-003	50.0	50.0	0.0	150	18
Vicuna-13B	41.89	58.11	0.0	249	19
Vicuna-7B	30.31	69.69	0.0	308	20
LLaMA2-13b-chat	9.06	90.94	0.0	1309	21

Table 1: The leaderboard for ChineseAlpacaEval. The 'Length' column is the average length of the model responses.

Baichuan-chat[6], Spark, Qwen, Chinese-alpaca-2-13b (Cui et al., 2023), Firefly-llama2-13b-chat (Yang, 2023), and BELLE-Llama2-13B-chat-0.4M (Yunjie Ji and Li, 2023). The results of the tests are presented in Table 1, referred to as the leaderboard.

Although LLaMA2-chat (Touvron et al., 2023) and Vicuna(Chiang et al., 2023) have a good ability to follow English instructions, the lack of Chinese-related data for training leads to poor Chinese ability. Among them, LLaMA2-chat can hardly generate Chinese properly when using the default prompt. Due to some Chinese closed-source LLMs having limitations on input and output, they can't generate responses for a small number of instructions. This part can only use the empty string as the corresponding response.

4 Limitations

Similar to many existing benchmarks that utilize LLMs for automated evaluation, ChineseAlpacaEval has some significant limitations, so it cannot substitute human evaluation in significant circumstances.

- ChineseAlpacaEval has not undergone detailed human consistency testing yet. Although AlpacaEval has demonstrated high consistency between automated evaluation using LLMs and human evaluation, this does not guarantee the same level of consistency between similar LLM-based automated evaluation and human evaluation in the Chinese context.

- ChineseAlpacaEval primarily evaluates the Chinese instruction-following capability of LLMs and their alignment with Chinese user preferences. It focuses on usefulness but cannot be used for evaluating safety aspects such as harmlessness and honesty.

- Due to limitations in evaluation time and cost, ChineseAlpacaEval has a limited scale for evaluating instruction sets, which means it cannot cover all real-world application scenarios. The instructions included may not fully represent actual usage scenarios.

- Automated evaluation with LLMs may not be able to identify all instances of hallucination phenomena in model generation.

[6]https://github.com/baichuan-inc/Baichuan-13B

References

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 90%* chatgpt quality.

Yiming Cui, Ziqing Yang, and Xin Yao. 2023. Efficient and effective text encoding for chinese llama and alpaca. *arXiv preprint arXiv:2304.08177*.

Zhengxiao Du, Yujie Qian, Xiao Liu, Ming Ding, Jiezhong Qiu, Zhilin Yang, and Jie Tang. 2022. Glm: General language model pretraining with autoregressive blank infilling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 320–335.

Yann Dubois, Xuechen Li, Rohan Taori, Tianyi Zhang, Ishaan Gulrajani, Jimmy Ba, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Alpacafarm: A simulation framework for methods that learn from human feedback.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. In *International Conference on Learning Representations*.

Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, Yao Fu, Maosong Sun, and Junxian He. 2023. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. *arXiv preprint arXiv:2305.08322*.

Haonan Li, Yixuan Zhang, Fajri Koto, Yifei Yang, Hai Zhao, Yeyun Gong, Nan Duan, and Timothy Baldwin. 2023a. Cmmlu: Measuring massive multitask language understanding in chinese.

Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023b. Alpacaeval: An automatic evaluator of instruction-following models. `https://github.com/tatsu-lab/alpaca_eval`.

OpenAI. 2023. Gpt-4 technical report.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Jianxin Yang. 2023. Firefly(): . `https://github.com/yangjianxin1/Firefly`.

Yan Gong Yiping Peng Qiang Niu Baochang Ma Yunjie Ji, Yong Deng and Xiangang Li. 2023. Belle: Be everyone's large language model engine. `https://github.com/LianjiaTech/BELLE`.

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. In *The Eleventh International Conference on Learning Representations*.

Xiaotian Zhang, Chunyang Li, Yi Zong, Zhengyu Ying, Liang He, and Xipeng Qiu. 2023. Evaluating the performance of large language models on gaokao benchmark.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. 2023. A survey of large language models. *arXiv preprint arXiv:2303.18223*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena.