A Systematic Evaluation of Large Language Models for Natural Language Generation Tasks

Xuanfan Ni, Piji Li*
College of Computer Science and Technology,
Nanjing University of Aeronautics and Astronautics
MIIT Key Laboratory of Pattern Analysis and Machine Intelligence
Nanjing, Jiangsu, 210016, China
{xuanfanni,pjli}@nuaa.edu.cn

Abstract

Recent efforts have evaluated large language models (LLMs) in areas such as commonsense reasoning, mathematical reasoning, and code generation. However, to the best of our knowledge, no work has specifically investigated the performance of LLMs in natural language generation (NLG) tasks, a pivotal criterion for determining model excellence. Thus, this paper conducts a comprehensive evaluation of well-known and high-performing LLMs, namely ChatGPT, ChatGLM, T5-based models, LLaMA-based models, and Pythia-based models, in the context of NLG tasks. We select English and Chinese datasets encompassing Dialogue Generation and Text Summarization. Moreover, we propose a common evaluation setting that incorporates input templates and post-processing strategies. Our study reports both automatic results, accompanied by a detailed analysis.

1 Introduction

Recent studies have emphasized the importance of scaling large language models (LLMs), referring to both the dimensions of the model size themselves and the amount of data used, resulting in enhanced capability of the models for tasks downstream (Chung et al., 2022). Numerous investigations have been conducted to explore the limits of performance by training increasingly larger pre-trained language models, such as GPT-3 175B (Brown et al., 2020) and PaLM 540B (Chowdhery et al., 2022). Although scaling primarily involves increasing the model size while maintaining similar architectures and pre-training tasks, these large-sized PLMs exhibit distinct behaviors from their smaller counterparts and demonstrate surprising **emergent abilities** in solving complex tasks (Zhang et al., 2017; Frankle and Carbin, 2019; Zhang et al., 2021). An example of this is the contrasting performance of GPT-3 and GPT-2 when it comes to solving few-shot tasks. GPT-3 demonstrates effective problem-solving abilities by utilizing in-context learning, whereas GPT-2 faces difficulties in this aspect. As a result, these large-scale language models (LLMs) has become a huge research topic in current NLP area. In existing literature, remarkable LLMs such as ChatGPT[0], ChatGLM[1], have been widely adopted as powerful AI

*Corresponding author.

[0]https://chat.openai.com/
[1]https://chatglm.cn/

assistants, benefiting from their exceptional generation capabilities.

We hypothesis that a language model's performance in executing natural language generation (NLG) tasks is a crucial factor in determining its excellence (Dong et al., 2023). NLG tasks involve LLMs that are capable of accepting diverse types of input, such as texts and tables, and generating coherent and appropriate output text. We intuitively think that generate fluent, coherent, and consistent texts is the foundation of a language model, so as to large language models (Raffel et al., 2020). When some research institutions release their large language models, they tend to evaluate these models first. Community workers are also interested in testing well-known large language models. However, most of these evaluations focus on checking LLMs' ability of commonsense reasoning (Davis and Marcus, 2015; Wei et al., 2022), mathematical reasoning (Saxton et al., 2019; Wei et al., 2022), code completion (Allamanis et al., 2018), etc., but ignore the basic NLG tasks, such as dialogue generation (Chen et al., 2017), text summarization (Dong et al., 2023), and story generation (Al-Hussain and Azmi, 2022). Besides, Some researchers pointed out that the performance of a large model is determined not only by its size and architecture, but more by the quality and quantity of training data. Based on this point of view, researchers open source and propose that some smaller-scale models trained on more and higher-quality data sets can achieve the same performance as models with more parameters than them. For example, LLaMA-13B (Touvron et al., 2023) outperforms GPT-3 on most benchmarks, despite being 10 times smaller. This notable discovery makes us curious about the performance of models with different architecture, data size, and mode size, trying to figure out which factor is more important. Therefore, we aim to address this gap by conducting a comparative analysis of LLM performance on NLG tasks, considering different architectures and scales throughout the evaluation process.

In this paper, we present a systematic evaluation of existing LLMs for NLG tasks. The main objective is to enhance our understanding of instruction and prompt design by conducting a comparative analysis of these models. Initially, we provide an overview of classic NLG tasks, including their definitions and associated English and Chinese datasets. Subsequently, we devise a model input template that includes instructions for each dataset. Following that, we introduce various LLMs, considering factors such as model size and architecture. Finally, we present the results of both automatic and manual evaluation of LLMs on NLG datasets, and discuss the strengths and weaknesses of their performance across different models.

2 Natural Language Generation

In this section, we will introduce the definition of NLG, and its sub-tasks with some corresponding datasets that we will use to evaluate LLMs.

2.1 Definition

Natural Language Generation is the process of producing a natural language text in order to meet specified communicative goals. The texts that are generated may range from a single phrase given in answer to a question, through multi-sentence remarks and questions within a dialog, to full-page explanations. In our evaluation, we mainly focus on text-to-text styles. In general, the task of NLG targets at finding an optimal sequence $y_{<T+1} = (y_1, y_2, \ldots, y_T)$ that satisfies:

$$y_{<T+1} = \underset{y_{<T+1} \in \mathcal{Y}}{\arg\max} \log P_\theta\left(y_{<T+1} \mid x\right) = \underset{y_{<T+1} \in \mathcal{Y}}{\arg\max} \sum_{t=1}^{T} \log P_\theta\left(y_t \mid y_{<t}, x\right) \tag{1}$$

where T represents the number of tokens of the generated sequence, $\mathcal{Y}$ represents a set containing all possible sequences, and $P_\theta\left(y_t \mid y_{<t}, x\right)$ is the conditional probability of the next token y_t based on its previous tokens $y_{<t} = (y_1, y_2, \ldots, y_{t-1})$ and the source sequence x with model parameters θ.

Next, we will introduce some classic and widely-researched sub-tasks of NLG, with several corresponding datasets.

2.2 Dialogue Generation

Dialogue generation refers to the process of automatically generating coherent and contextually appropriate responses in a conversational setting (Chen et al., 2017; Ma et al., 2020; Dong et al., 2023). The ultimate goal of dialogue generation task is to create responses that are relevant, informative, and engaging to the user.We utilize two English dialogue datasets characterized by clear emotional flow and topic constraints, as well as one English dataset that incorporates speakers' personalities. Furthermore, we employ a Chinese open-domain dialogue dataset for evaluation purposes.

- **DailyDialog** (Li et al., 2017) is a comprehensive, human-authored, and relatively noise-free English dataset that captures everyday communication styles and encompasses various topics related to our daily lives.
- **PersonaChat** (Zhang et al., 2018) is a persona-grounded dialogue dataset which contains 10k English multi-turn dialogues conditioned on personas, and each persona is described with at least 5 profile sentences.
- **EmpatheticDialogue** (Rashkin et al., 2019) is a large-scale multi-turn dialogue English dataset that contains 25k empathetic conversations between a speaker and a listener.
- **LCCC** (Wang et al., 2020) is a large-scale cleaned Chinese conversation dataset.

2.3 Text Summarization

Text summarization is the process of condensing a piece of text, such as an article, document, or news story, into a shorter version while preserving its key information and main ideas (El-Kassas et al., 2021; Dong et al., 2023). Text summarization can be performed through two main approaches: *Extractive Summarization* and *Abstractive Summarization*. In our evaluation, we utilize multiple abstractive summarization datasets, specifically choosing two renowned datasets for the English and Chinese languages.

- **CNN/DailyMail** (Nallapati et al., 2016) is a large scale English summarization dataset which contains 93k and 220k articles collected from the CNN and Daily Mail websites, respectively, where each article has its matching abstractive summary.
- **XSum** (Narayan et al., 2018) is an extreme English summarization dataset containing BBC articles and corresponding single sentence summaries. In this dataset, 226,711 Wayback archived BBC articles are collected, which range from 2010 to 2017 and cover a wide variety of domains.

- **THUCNews** (Li and Sun, 2007) is a Chinese summarization dataset, which comes from filtering the historical data of the Sina News RSS subscription channel from 2005 to 2011, including 740,000 news documents.
- **LCSTS** (Liu, 2020) is a large corpus of Chinese short text summarization dataset constructed from the Chinese micro-blogging website *Sina Weibo*. This corpus consists of over 2 million real Chinese short texts with short summaries given by the author of each text.

2.4 Overview for LLMs

Typically, large language models (LLMs) refer to Transformer-based models containing tens or hundreds of billions of parameters and trained on extensive corpora of texts (Zhao et al., 2023). These LLMs demonstrate significant capabilities in understanding natural language and solving complex tasks. Furthermore, they have showcased their ability to perform new tasks based on textual instructions or with just a few examples (Chung et al., 2022). The emergence of these few-shot properties is a result of scaling models to a sufficient size, leading to a line of research that focuses on further scaling these models (Rae et al., 2021).

Previous LLMs, such as T5 (Raffel et al., 2020), GPT-3 (Brown et al., 2020), OPT (Zhang et al., 2022), and PaLM (Chowdhery et al., 2022), primarily emphasized scaling model size rather than considering the quality and quantity of data. However, recent studies have demonstrated that, given a fixed compute budget, the best performance is achieved by smaller models trained on larger datasets (Hoffmann et al., 2022). Additionally, most of these models are not open-source and can only be accessed through APIs for inference, which poses inconveniences for model evaluation and usage. In order to address this issue, numerous researchers have proposed excellent open-source architectures and trained models, including GLM-130B (Zeng et al., 2022), ChatGLM (Du et al., 2022), LLaMA (Touvron et al., 2023), and Pythia (Biderman et al., 2023). Furthermore, advancements in fine-tuning techniques have contributed to the success of deploying these models with limited resources, such as Lora (Hu et al., 2022) and P-Tuning (Li and Liang, 2021). Therefore, this paper aims to conduct systematic evaluations of these models and their fine-tuned versions, categorized into four groups: **ChatGPT, ChatGLM, T5-based models, LLaMA-based models, and Pythia-based models**.

2.5 ChatGPT

ChatGPT[2] is a large language model based on OpenAI's GPT-3.5 architecture (Brown et al., 2020). It is designed specifically for generating conversations and answering user queries. ChatGPT employs large-scale pretraining and fine-tuning methodologies, utilizing vast amounts of textual data to learn statistical patterns and semantic knowledge of language, and perform well in zero-shot and few-shot settings, and can understand the input instructions.

2.6 ChatGLM

ChatGLM[3] is a freely available dialogue language model that operates in both Chinese and English languages. It follows the GLM architecture and boasts an impressive parameter count of 6.2 billion. ChatGLM-6B incorporates similar technology as ChatGPT, with a specific focus

[2]https://chat.openai.com/
[3]https://chatglm.cn/

on Chinese question answering and dialogue. The model undergoes extensive training on a dataset containing approximately 1 trillion tokens in Chinese and English. The training process includes supervised fine-tuning, feedback bootstrap, and reinforcement learning with human feedback. Despite having only 6.2 billion parameters, the model demonstrates the ability to generate responses that align with human preferences.

2.7 T5-Based models

T5 (Raffel et al., 2020), which stands for Text-To-Text Transfer Transformer, is a transformer-based language model developed by Google Research. Instead of training separate models for different tasks, T5 is trained in a text-to-text pattern. This means that it is trained to perform a wide range of NLP tasks by transforming the input text into a standardized format that specifies the task to be performed. In our evaluation, we select two new fine-tuned versions of T5, namely: Flan-T5-XXL[4] and FastChat-T5[5].

Flan-T5-XXL Flan-T5 (Chung et al., 2022) is a fine-tuned version model class of T5 that has been trained on a variety of datasets phrased as instructions. It has shown impressive performance on several benchmarks, demonstrating strong zero-shot, few-shot, and Chain-of-Thought (CoT) (Wei et al., 2022) abilities. Flan-T5-XXL is the largest released checkpoint of this model, boasting a parameter volume of 13B. It inherits the extensive knowledge base of T5 while also being capable of understanding natural language instructions and performing the corresponding tasks.

FastChat-T5 FastChat (Zheng et al., 2023a) is an open platform for training, serving, and evaluating large language model based chatbots. And FastChat-T5 is an open-source chatbot trained on this platform by fine-tuning Flan-T5-XL (3B parameters) on user-shared conversations collected from ShareGPT.

2.8 LLaMA-Based Models

LLaMA (Touvron et al., 2023) is a collection of foundation language models ranging from 7B to 65B parameters proposed by Meta AI. Unlike other famous LLMs, LLaMA is only trained on publicly avaiable data, making it compatible with open-sourcing. Numerous remarkable and impressive models have emerged as a result, built upon the LLaMA framework and trained using diverse datasets. Among these models, we have chosen a few prominent ones for evaluation: Open-LLaMA, Vicuna, Alpaca, and GPT4ALL.

Open-LLaMA Open-LLaMA (Geng and Liu, 2023) is an open reproduction of LLaMA trained on the RedPajama dataset (Computer, 2023). We leverage the 7B version[6] of this model for evaluation.

Alpaca (Taori et al., 2023) is fine-tuned based on a 7B LLaMA model using a dataset consisting of 52,000 instances of instruction-following data. This dataset is generated using the techniques outlined in the Self-Instruct paper (Wang et al., 2022), which aims to address the limited instruction-following capabilities of LLaMA models. To create the training data, the

[4]https://huggingface.co/google/flan-t5-xxl
[5]https://huggingface.co/lmsys/fastchat-t5-3b-v1.0
[6]https://github.com/openlm-research/open_llama

authors initially generate the data using OpenAI's GPT-3 and subsequently convert it into 52,000 instances of instruction-following conversational data using the Self-Instruct pipeline. This dataset is referred to as the Alpaca dataset. The Alpaca model is then fine-tuned to generate responses in conversations similar to ChatGPT.

In our evaluation, we utilize Alpaca-Lora-7B[7], a low-rank adapter for LLaMA-7b fit on the Stanford Alpaca dataset, and Chinese-Alpaca-13b[8], a Chinese model version of Alpaca.

Vicuna (Zheng et al., 2023b) is fine-tuned based on LLaMA models using user-shared conversations collected from ShareGPT. It is an auto-regressive language model, based on the transformer architecture. So it is basically fine-tuned with ChatGPT conversations. We utilize the 13B version of Vicuna, which is Vicuna-13B[9].

GPT4ALL (Anand et al., 2023) is a fine-tuned LLaMA 13B model and the GPT4All community[10] has built the GPT4All Open Source datalake as a staging ground for contributing instruction and assistant tuning data for future GPT4All model trains.

2.9 Pythia-Based Models

Pythia (Biderman et al., 2023) is a project by EleutherAI[11] that combines interpret-ability analysis and scaling laws to understand how knowledge develops and evolves during training in autoregressive Transformers. We utilize two versions of Pythia which are Oasst-Pythia and Dolly.

Oasst-Pythia[12] is an open assistant model developed by the Open-Assistant project. It is based on a Pythia 12B model that was fine-tuned on human demonstrations of assistant conversations collected through the Open-Assistant human feedback web app.

Dolly[13] is a Language Model (LLM) with 12B parameters, designed to follow instructions accurately. It has been trained on approximately 15,000 instruction/response fine-tuning records known as databricks-dolly-15k. These records were created by Databricks employees and cover various capability domains sourced from InstructGPT (Ouyang et al., 2022). These domains include brainstorming, classification, closed QA, generation, information extraction, open QA, and summarization.

3 Experimental Settings

3.1 Dataset

In our evaluation, we aim to showcase the generation capabilities of LLMs in zero-shot scenarios. Therefore, we refrain from providing any additional information to the model for each of the aforementioned datasets. Specifically:

- For datasets of Text Summarization task, we input the text, document, or article to allow the model to extract key information and generate concise summaries.

[7] https://huggingface.co/chainyo/alpaca-lora-7b
[8] https://huggingface.co/shibing624/chinese-alpaca-plus-13b-hf
[9] https://huggingface.co/eachadea/vicuna-13b-1.1
[10] https://home.nomic.ai/
[11] https://github.com/EleutherAI/pythia
[12] https://huggingface.co/OpenAssistant/pythia-12b-sft-v8-7k-steps
[13] https://huggingface.co/databricks/dolly-v2-12b

```
Below is an instruction that describes
a task. Write a response that appropr-
iately completes the request.
### Instruction: {instruction}
### Input: {text}
### Response:
```

```
以下是描述任务的说明。 编写准确的回复来
完成这个任务。

### 说明: {instruction}
### 输入: {text}
### 回复:
```

Figure 1: Input templates for English (left) and Chinese (right) datasets. **instruction** and **text** will be replaced with content corresponding different datasets.

Dataset	Instruction	Text
Empathetic Dialogues	This is an open-domain *empathetic* dialogue completion task.The input is the Dialogue. You act as System in the dialogue. You need to fully *understand the situation and combine the speaker's emotion* to complete the dialogue with natural content and a way closer to human speech. There is no need for any additional notes or clarifications, you just give the response in English.	Dialogue Context
DailyDialog	This is an open-domain *topic-aware* dialogue completion task. The input is the Dialogue. You act as System in the dialogue. You need to fully *understand the topic* and complete the dialogue with natural content and a way closer to human speech. There is no need for any additional notes or clarifications, you just give the response in English	Dialogue Context
PersonaChat	This is an open-domain *personality-aware* dialogue completion task. The input is the Dialogue. You act as System in the dialogue. You need to fully *understand the personality* and complete the dialogue with natural content and a way closer to human speech. There is no need for any additional notes or clarifications, you just give the response in English.	Dialogue Context
LCCC	这是一个开放域的中文对话补全任务。输入是待完成的对话内容。你在对话中扮演系统。你需要完全理解说话者的话语，并用自然的内容和更接近于人类说话的方式完成对话，而不是用语言模型或者AI的身份。不需要任何额外的注释或者说明，你只需用中文给出回复。	Dialogue Context

Table 1: *Instruction* and *Text* for each dataset.

- For datasets of Dialogue Generation task, we input the dialogue history, enabling the model to generate appropriate responses for the final round of the conversation.

We defer the evaluation of LLMs on Chinese datasets and other NLG tasks such as story generation, along with results of manual and GPT-4 rating, to future research endeavors.

3.2 Input Template

Because LLMs that we evaluate possess the ability to comprehend instructions and perform corresponding tasks, so in order to ensure fairness, we develop an input template that is applied to every dataset for each task, serving as the input for every large language model. This template consists of two components: the instruction and the input. Figure 1 illustrates the templates designed for both the Chinese and English datasets, and Table 1 shows the content of *instruction* and *text* for each dataset.

3.3 Hyperparameters

Although each LLM may have its own optimal decoding strategy, for the sake of fairness, we have standardized these hyperparameters across all LLMs. We employ the Top-k and Top-p sampling, with $k = 40$ and $p = 0.75$. Additionally, a temperature value of 0.2 and a repetition penalty factor of 1.15 are imposed. Furthermore, we specify a maximum token length of 512 and a minimum token length of 10 for the generated content.

3.4 Post-Processing Strategy

Through case study, we observe that despite emphasizing the exclusion of any additional output in the input, regrettably, most LLMs still generate redundant information in their output. Therefore, we find it necessary to apply post-processing to the outputs of these models. To ensure fairness, we adopt the same post-processing strategy for all LLMs. Specifically, we utilize the keywords "`### response:`" or "`### 回复：`" for segmentation. If the segmented content consists of a single line, we consider it as the final result. If the segmented content spans multiple lines, we use "`\n`" as segmentation keywords and select the first sentence with a length not less than 16 as the final result.

3.5 Baselines

There have been numerous previous works on datasets we used, and these works have achieved good results. Therefore, despite the fact that most of these works have proposed models much smaller than LLMs and have predominantly utilized supervised fine-tuning methods, we still compare them with LLMs to highlight some characteristics of LLMs. For each dataset, we select several recent works with better performance and report their results.

- For EmpatheticDialogues, we utilize **EP-PG** (Li et al., 2022) that first generates event transition plans and then obtains the final response, and **MoEL** (Lin et al., 2019) that are consist of one emotion tracker and n emotion listeners.
- For DailyDialog, we utilize **PLATO** (Bao et al., 2020), a pre-trained dialogue generation model, and **DialogWAE** (Gu et al., 2019), a conditional wasserstein autoencoder (WAE) specially designed for dialogue modeling.
- For PersonaChat, we utilize **PLATO** as mentioned above, and **CTRLStruct** (Yin et al., 2023) for dialogue structure learning to effectively explore topic-level dialogue clusters.

3.6 Evaluation Metrics

Automatic Metrics We utilize several common automatic metrics for NLG tasks. **PPL** is used to assess the difficulty or confusion of a language model in predicting a sequence of words. **BLEU** (B-1, B-2, B-3, B-4) (Papineni et al., 2002) is used to assess the quality of machine-generated translations by comparing them to human reference translations. **Meteor** (MT) (Banerjee and Lavie, 2005) considers the accuracy and recall based on the entire corpus, and get the final measure. **Rouge-L** (R-L) (Lin, 2004) calculates the overlap between the generated output and the reference summaries or translations using various techniques such as N-gram matching.

Model	**Scale**	**Arch**	**PPL↓**	**B-1**	**B-2**	**B-4**	**MT**	**R-L**	**D-1**	**D-2**	**PPR↓**
EP-PG	–	–	–	16.74	6.94	2.39	–	–	2.19	8.25	–
MoEL	23.1M	DO	33.58	–	–	2.90	–	–	1.06	4.29	–
ChatGPT	175B	DO	10.52	7.35	2.40	0.52	**9.26**	**8.75**	4.71	27.75	**0.00%**
ChatGLM	6B	DO	11.73	6.05	1.82	0.27	8.58	7.71	3.57	22.82	12.61%
Flan-T5-XXL	13B	ED	19.97	5.62	2.40	<u>0.61</u>	5.38	7.41	5.66	24.97	**0.00%**
FastChat-T5	3B	ED	<u>9.25</u>	7.33	2.35	0.45	8.50	8.62	3.55	20.81	0.12%
Open-LLaMA	7B	DO	15.90	**8.50**	**2.97**	**0.63**	6.43	<u>8.74</u>	3.93	17.91	40.05%
Vicuna	13B	DO	14.31	6.18	1.93	0.35	<u>8.91</u>	7.81	4.09	25.84	38.86%
Alpaca-Lora	7B	DO	16.10	7.95	2.52	0.40	7.34	6.69	**7.59**	<u>39.58</u>	0.24%
Chinese-Alpaca	13B	DO	12.05	6.51	1.86	0.35	7.53	6.64	5.32	29.14	0.20%
GPT4ALL	13B	DO	11.14	5.20	1.47	0.24	8.75	6.78	3.94	25.60	1.81%
Dolly	12B	DO	131.75	<u>8.29</u>	<u>2.64</u>	0.46	6.91	7.96	<u>7.46</u>	**42.69**	58.61%
Oasst-Pythia	12B	DO	**8.71**	5.48	1.53	0.26	8.79	6.92	3.38	21.18	<u>0.04%</u>

Table 2: Automatic evaluation results of LLMs on EmpatheticDialogues. **Scale** stands for the model size.**ED** and **DO** respectively stand for *encoder-decoder* and *decoder-only*. **Arch** is an abbreviation for *Architecture*. The **bold** numbers in the results represent the best scores, whereas the <u>underlined</u> numbers indicate the second-best scores.

DISTINCT (D-1, D-2) (Li et al., 2016) quantifies how many distinct or different N-grams are present in the generated text, providing an indication of the model's ability to produce varied and non-repetitive output.

Besides these widely-used metrics, we also develop a new metric called **PostProcess Rate** (PPR), which means the proportion of samples that need to be post-processed to the total number of samples.

4 Results and Analysis

4.1 Dialogue Generation

The automatic metrics results of LLMs on the three datasets are shown in Tables 2, 3, and 4. Although automatic metrics cannot fully reflect the performance of the models, we can still draw the following conclusions from them.

First, apart from ChatGPT that has the largest scale of 175B, the two T5-based models consistently outperform others in terms of the **PPR** metric. This indicates that the generated content of Flan-T5-XXL and FastChat-T5 largely aligns with the instruction requirements stated in the input template: "*without any additional output.*" Interestingly, both of these models follow an encoder-decoder architecture, while all other models follow a decoder-only architecture. This suggests that encoder-decoder models demonstrate superior understanding of input instructions under the same model scale. We speculate that having an encoder allows the model to comprehend the input content effectively, thereby executing the corresponding task more successfully.

Second, Alpaca-Lora consistently ranks either first or second in the richness of output content. Moreover, the models using the same architecture as Alpaca-Lora also achieve higher scores in

Model	Scale	Arch	PPL↓	B-1	B-2	B-4	MT	R-L	D-1	D-2	PPR↓
PLATO	–	DO	–	39.70	31.10	–	–	–	5.30	29.10	–
DialogWAE	–	ED	–	32.30	–	–	–	–	31.30	59.70	–
ChatGPT	175B	DO	11.41	7.58	2.71	0.56	**10.13**	8.17	10.98	47.20	**0.00%**
ChatGLM	6B	DO	17.52	**10.54**	**3.86**	**0.93**	9.14	**11.91**	9.60	42.69	12.05%
Flan-T5-XXL	13B	ED	16.31	3.85	1.61	0.42	6.64	5.52	14.54	47.59	**0.00%**
FastChat-T5	3B	ED	**10.27**	7.45	2.59	0.50	9.15	7.86	9.58	41.16	0.50%
Open-LLaMA	7B	DO	21.23	6.72	2.31	0.46	5.94	5.59	11.65	38.72	64.36%
Vicuna	13B	DO	78.66	6.13	2.11	0.42	8.89	6.96	10.15	45.18	38.55%
Alpaca-Lora	7B	DO	28.63	6.40	2.16	0.00	6.04	5.02	**17.49**	**61.66**	3.41%
Chinese-Alpaca	13B	DO	22.23	6.52	2.18	0.38	7.49	5.93	13.06	51.02	2.01%
GPT4ALL	13B	DO	14.72	4.84	1.24	0.13	7.72	5.77	10.24	43.53	25.50%
Dolly	12B	DO	58.29	6.09	2.01	0.40	5.70	4.25	14.14	52.33	74.80%
Oasst-Pythia	12B	DO	10.68	5.40	1.45	0.19	7.62	6.09	9.23	38.91	16.47%

Table 3: Automatic evaluation results of LLMs on DailyDialog.

Model	Scale	Arch	PPL↓	B-1	B-2	B-4	MT	R-L	D-1	D-2	PPR↓
PLATO	–	DO	–	40.60	31.50	–	–	–	2.10	12.10	–
CTRLStruct	–	ED	–	31.60	11.90	–	–	16.10	3.20	11.40	–
ChatGPT	175B	DO	10.97	6.36	2.37	**0.52**	**9.78**	8.42	9.10	40.65	**0.00%**
ChatGLM	6B	DO	13.89	5.98	2.07	0.40	8.85	**8.67**	6.85	34.86	12.05%
Flan-T5-XXL	13B	ED	51.50	**6.51**	**2.53**	0.43	6.15	7.46	12.23	39.82	**0.00%**
FastChat-T5	3B	ED	10.61	5.53	2.00	0.43	8.98	7.94	7.30	33.66	0.50%
Open-LLaMA	7B	DO	15.69	4.43	1.16	0.00	5.86	5.43	7.83	28.90	64.36%
Vicuna	13B	DO	12.53	3.20	1.01	0.14	7.30	4.82	5.88	30.12	38.55%
Alpaca-Lora	7B	DO	17.20	4.19	1.21	0.24	6.29	4.40	**12.28**	**50.33**	3.41%
Chinese-Alpaca	13B	DO	14.95	4.93	1.66	0.29	7.70	6.21	10.18	44.62	2.01%
GPT4ALL	13B	DO	11.68	2.74	0.55	0.07	6.52	4.39	7.56	35.23	25.50%
Dolly	12B	DO	29.76	4.51	1.39	0.24	5.02	4.59	10.55	41.62	74.80%
Oasst-Pythia	12B	DO	**9.57**	3.34	0.69	0.07	6.58	4.66	6.48	28.56	16.47%

Table 4: Automatic evaluation results of LLMs on PersonaChat.

terms of D-1 and D-2. This indicates that LLAMA-based models are capable of producing more diverse and less repetitive content.

Last, ChatGPT, the model with the largest parameter scale, performs the best overall on all four datasets, securing the first or second position most frequently. This suggests that increasing the parameter size and training data volume of LLMs is consistently one of the most important methods for improving model performance.

4.2 Text Summarization

The automatic metrics results of LLMs on the three datasets are shown in Tables 5 and 6. In addition to the widely used Rouge-L in text summarization tasks, we have incorporated several

Model	**Scale**	**Arch**	**PPL↓**	**B-1**	**B-2**	**B-3**	**B-4**	**MT**	**R-L**	**PPR↓**
ChatGPT	175B	DO	<u>10.86</u>	2.99	0.58	0.00	0.00	4.89	5.02	**0.00%**
ChatGLM	6B	DO	18.56	2.80	0.87	0.25	0.00	4.80	4.91	10.78%
Flan-T5-XXL	13B	ED	15.96	**5.49**	<u>1.21</u>	0.00	0.00	3.69	5.16	**0.00%**
FastChat-T5	3B	ED	**10.26**	2.62	0.89	0.46	0.29	4.80	4.58	<u>0.03%</u>
Open-LLaMA	7B	DO	45.72	0.02	0.01	0.00	0.00	0.35	0.18	73.67%
Vicuna	13B	DO	10.94	2.45	0.81	0.41	0.23	4.75	4.40	31.29%
Alpaca-lora	7B	DO	19.22	3.41	0.56	0.00	0.00	4.19	4.23	0.13%
Chinese-Alpaca	13B	DO	14.30	<u>4.40</u>	**1.88**	**1.13**	**0.74**	3.55	**10.27**	0.15%
GPT4ALL	13B	DO	23.28	3.03	0.85	0.49	0.35	5.14	5.06	3.37%
Dolly	12B	DO	15.01	3.35	1.12	<u>0.62</u>	0.40	**5.40**	6.01	46.67%
Oasst-Pythia	12B	DO	18.83	3.48	1.15	0.61	<u>0.41</u>	<u>5.23</u>	<u>6.31</u>	0.08%

Table 5: Automatic evaluation results of LLMs on CNN/DailyMail.

Model	**Scale**	**Arch**	**PPL↓**	**B-1**	**B-2**	**B-3**	**B-4**	**MT**	**R-L**	**PPR↓**
ChatGPT	175B	DO	14.92	7.55	2.93	1.27	0.55	11.47	10.31	**0.00%**
ChatGLM	6B	DO	22.84	5.45	2.46	1.19	0.60	10.76	9.25	8.79%
Flan-T5-XXL	13B	ED	**10.90**	**12.48**	**4.66**	**2.19**	**1.81**	**17.60**	**15.06**	**0.00%**
FastChat-T5	3B	ED	<u>14.08</u>	8.05	<u>3.78</u>	1.83	0.78	<u>13.22</u>	<u>11.01</u>	**0.00%**
Open-LLaMA	7B	DO	31.13	4.57	1.31	0.55	0.00	2.31	2.70	56.79%
Vicuna	13B	DO	14.58	7.13	3.06	1.41	0.67	12.61	10.16	30.11%
Alpaca-lora	7B	DO	23.49	<u>8.65</u>	2.95	1.20	0.49	10.94	9.54	<u>1.17%</u>
Chinese-Alpaca	13B	DO	19.21	6.65	3.31	<u>1.88</u>	<u>1.19</u>	5.98	8.34	5.90%
GPT4ALL	13B	DO	18.79	8.47	3.46	1.68	0.95	11.73	9.81	15.79%
Dolly	12B	DO	20.89	6.44	2.64	1.01	0.00	11.21	9.95	82.23%
Oasst-Pythia	12B	DO	21.49	6.27	2.46	0.99	0.37	9.98	9.32	28.42%

Table 6: Automatic evaluation results of LLMs on XSum.

other metrics, such as BLEU and PPL, to enhance the diversity of the results. Our observations from the two datasets can be summarized as follows:

The Flan-T5 and FastChat-T5 models employ an encoder-decoder architecture, exhibiting remarkable proficiency in instruction comprehension, as evident by their minimal requirement for post-processing. This finding is corroborated by the analysis of dialogue generation. Moreover, our investigation on the XSum dataset reveals that both models surpass other LLMs, consistently attaining top positions across various metrics such as BLEU and ROUGE scores. These impressive results are likely attributed to the inherent strengths embedded within their model structures.

5 Conclusion

In this paper, we conduct a comprehensive assessment of several existing large-scale language models (LLMs) in the context of natural language generation (NLG) tasks. Our evaluation encompasses English and Chinese datasets to examine the multilingual capabilities of these LLMs. The results and analyses from both automatic and manual evaluations of LLMs reveal notable trends and phenomena.

Acknowledgements

This research is supported by the National Natural Science Foundation of China (No.62106105), the CCF-Tencent Open Research Fund (No.RAGR20220122), the CCF-Zhipu AI Large Model Fund (No.CCF-Zhipu202315), the Scientific Research Starting Foundation of Nanjing University of Aeronautics and Astronautics (No.YQR21022), and the High Performance Computing Platform of Nanjing University of Aeronautics and Astronautics.

References

Arwa Al-Hussain and Aqil M. Azmi. 2022. Automatic story generation: A survey of approaches. ACM Comput. Surv., 54(5):103:1–103:38.

Miltiadis Allamanis, Earl T. Barr, Premkumar T. Devanbu, and Charles Sutton. 2018. A survey of machine learning for big code and naturalness. ACM Comput. Surv., 51(4):81:1–81:37.

Yuvanesh Anand, Zach Nussbaum, Brandon Duderstadt, Benjamin Schmidt, and Andriy Mulyar. 2023. Gpt4all: Training an assistant-style chatbot with large scale data distillation from gpt-3.5-turbo. https://github.com/nomic-ai/gpt4all.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In Jade Goldstein, Alon Lavie, Chin-Yew Lin, and Clare R. Voss, editors, Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization@ACL 2005, Ann Arbor, Michigan, USA, June 29, 2005, pages 65–72. Association for Computational Linguistics.

Siqi Bao, Huang He, Fan Wang, Hua Wu, and Haifeng Wang. 2020. PLATO: pre-trained dialogue generation model with discrete latent variable. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020, pages 85–96. Association for Computational Linguistics.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. CoRR, abs/2304.01373.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. CoRR, abs/2005.14165.

Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. SIGKDD Explor., 19(2):25–35.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways. CoRR, abs/2204.02311.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. CoRR, abs/2210.11416.

Together Computer. 2023. Redpajama-data: An open source recipe to reproduce llama training dataset.

Ernest Davis and Gary Marcus. 2015. Commonsense reasoning and commonsense knowledge in artificial intelligence. Commun. ACM, 58(9):92–103.

Chenhe Dong, Yinghui Li, Haifan Gong, Miaoxin Chen, Junxin Li, Ying Shen, and Min Yang. 2023. A survey of natural language generation. ACM Comput. Surv., 55(8):173:1–173:38.

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 Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022, pages 320–335. Association for Computational Linguistics.

Wafaa S. El-Kassas, Cherif R. Salama, Ahmed A. Rafea, and Hoda K. Mohamed. 2021. Automatic text summarization: A comprehensive survey. Expert Syst. Appl., 165:113679.

Jonathan Frankle and Michael Carbin. 2019. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net.

Xinyang Geng and Hao Liu. 2023. Openllama: An open reproduction of llama, May.

Xiaodong Gu, Kyunghyun Cho, Jung-Woo Ha, and Sunghun Kim. 2019. Dialogwae: Multimodal response generation with conditional wasserstein auto-encoder. In 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, Jack W. Rae, Oriol Vinyals, and Laurent Sifre. 2022. Training compute-optimal large language models. CoRR, abs/2203.15556.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022. OpenReview.net.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. In Chengqing Zong, Fei Xia, Wenjie Li, and Roberto Navigli, editors, Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021, pages 4582–4597. Association for Computational Linguistics.

Jingyang Li and Maosong Sun. 2007. Scalable term selection for text categorization. In Jason Eisner, editor, EMNLP-CoNLL 2007, Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, June 28-30, 2007, Prague, Czech Republic, pages 774–782. ACL.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In Kevin Knight, Ani Nenkova, and Owen Rambow, editors, NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016, pages 110–119. The Association for Computational Linguistics.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. Dailydialog: A manually labelled multi-turn dialogue dataset. In Greg Kondrak and Taro Watanabe, editors, Proceedings of the Eighth International Joint Conference on Natural Language Processing, IJCNLP 2017, Taipei, Taiwan, November 27 - December 1, 2017 - Volume 1: Long Papers, pages 986–995. Asian Federation of Natural Language Processing.

Qintong Li, Piji Li, Wei Bi, Zhaochun Ren, Yuxuan Lai, and Lingpeng Kong. 2022. Event transition planning for open-ended text generation. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022, pages 3412–3426. Association for Computational Linguistics.

Zhaojiang Lin, Andrea Madotto, Jamin Shin, Peng Xu, and Pascale Fung. 2019. Moel: Mixture of empathetic listeners. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019, pages 121–132. Association for Computational Linguistics.

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

Cong Liu. 2020. Chinese newstitle generation project by gpt2.

Yukun Ma, Khanh Linh Nguyen, Frank Z. Xing, and Erik Cambria. 2020. A survey on empathetic dialogue systems. Inf. Fusion, 64:50–70.

Ramesh Nallapati, Bowen Zhou, Cícero Nogueira dos Santos, Çaglar Gülçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. In Yoav Goldberg and Stefan Riezler, editors, Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016, pages 280–290. ACL.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018, pages 1797–1807. Association for Computational Linguistics.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F. Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In NeurIPS.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA, pages 311–318. ACL.

Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, H. Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu, Erich Elsen, Siddhant M. Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew J. Johnson, Blake A. Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Edward Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorrayne Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. 2021. Scaling language models: Methods, analysis & insights from training gopher. CoRR, abs/2112.11446.

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. J. Mach. Learn. Res., 21:140:1–140:67.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers, pages 5370–5381. Association for Computational Linguistics.

David Saxton, Edward Grefenstette, Felix Hill, and Pushmeet Kohli. 2019. Analysing mathematical reasoning abilities of neural models. In 7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net.

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.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurélien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. CoRR, abs/2302.13971.

Yida Wang, Pei Ke, Yinhe Zheng, Kaili Huang, Yong Jiang, Xiaoyan Zhu, and Minlie Huang. 2020. A large-scale chinese short-text conversation dataset. In Xiaodan Zhu, Min Zhang, Yu Hong, and Ruifang He, editors, Natural Language Processing and Chinese Computing - 9th CCF International Conference, NLPCC 2020, Zhengzhou, China, October 14-18, 2020, Proceedings, Part I, volume 12430 of Lecture Notes in Computer Science, pages 91–103. Springer.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. Self-instruct: Aligning language model with self generated instructions. CoRR, abs/2212.10560.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In NeurIPS.

Congchi Yin, Piji Li, and Zhaochun Ren. 2023. Ctrlstruct: Dialogue structure learning for open-domain response generation. In Ying Ding, Jie Tang, Juan F. Sequeda, Lora Aroyo, Carlos Castillo, and Geert-Jan Houben, editors, Proceedings of the ACM Web Conference 2023, WWW 2023, Austin, TX, USA, 30 April 2023 - 4 May 2023, pages 1539–1550. ACM.

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, Peng Zhang, Yuxiao Dong, and Jie Tang. 2022. GLM-130B: an open bilingual pre-trained model. CoRR, abs/2210.02414.

Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. 2017. Understanding deep learning requires rethinking generalization. In 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings. OpenReview.net.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? In Iryna Gurevych and Yusuke Miyao, editors, Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers, pages 2204–2213. Association for Computational Linguistics.

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

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona T. Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. OPT: open pre-trained transformer language models. CoRR, abs/2205.01068.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2023. A survey of large language models. CoRR, abs/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. 2023a. Judging llm-as-a-judge with mt-bench and chatbot arena. CoRR, abs/2306.05685.

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. 2023b. Judging llm-as-a-judge with mt-bench and chatbot arena.