SeqGPT: An Out-of-the-Box Large Language Model for Open Domain Sequence Understanding

Tianyu Yu[1*], Chengyue Jiang[2*], Chao Lou[2*], Shen Huang[4*]
Xiaobin Wang[4], Wei Liu[2], Jiong Cai[2], Yangning Li[1], Yinghui Li[1], Kewei Tu[2]
Hai-Tao Zheng[1], Ningyu Zhang[3], Pengjun Xie[4], Fei Huang[4], Yong Jiang[4†]

[1]Tsinghua University
[2]ShanghaiTech University
[3] Zhejiang University
[4]DAMO Academy, Alibaba Group
yiranytianyu@gmail.com {jiangchy,louchao}@shanghaitech.edu.cn
{pangda,xuanjie.wxb,yongjiang.jy}@alibaba-inc.com

Abstract

Large language models (LLMs) have shown impressive abilities for open-domain NLP tasks. However, LLMs are sometimes too footloose for natural language understanding (NLU) tasks which always have restricted output and input format. Their performances on NLU tasks are highly related to prompts or demonstrations and are shown to be poor at performing several representative NLU tasks, such as event extraction and entity typing. To this end, we present SeqGPT, a bilingual (i.e., English and Chinese) open-source autoregressive model specially enhanced for open-domain natural language understanding. We express all NLU tasks with two atomic tasks, which define fixed instructions to restrict the input and output format but still "open" for arbitrarily varied label sets. The model is first instruction-tuned with extremely fine-grained labeled data synthesized by ChatGPT and then further fine-tuned by 233 different atomic tasks from 152 datasets across various domains. The experimental results show that SeqGPT has decent classification and extraction ability, and is capable of performing language understanding tasks on unseen domains. We also conduct empirical studies on the scaling of data and model size as well as on the transfer across tasks. Our models are accessible at https://github.com/Alibaba-NLP/SeqGPT.

1 Introduction

Recent advancements in large language models (LLMs) have demonstrated their impressive ability across various NLP tasks (Kaplan et al. 2020; Wei et al. 2022b; Chung et al. 2022; Li et al. 2023c,d,b). Regarding natural language understanding (NLU) tasks, although the next-word-prediction approach utilized by language models implies little bias to the task-specific output structures, such as spans in named entity recognition (NER) and triplets in relation extraction (RE), numerous attempts (Qin et al. 2023; Wei et al. 2023; Wadhwa, Amir, and Wallace 2023; Ashok and Lipton 2023) have been made to apply LLMs to open-domain NLU tasks through the application of prompt engineering, mainly due to the LLMs' exceptional ability of generalization and instruction-following (Figure 1). However, the direct application of LLMs comes with notable drawbacks. Instruction-following necessitates the use of a sufficiently large model (Kaplan et al. 2020; Wei et al. 2022b), for example, GPT-3 (Brown et al. 2020) has 175B parameters, which can lead to considerable inference costs and challenges in customization (Hu et al. 2022; Liu et al. 2022a,b). In addition, prompt engineering is crucial to achieve promising performance and ensure adherence to output format standards. However, it is highly empirical and the models may not consistently abide by it (Chase 2022; Gravitas 2023).

To perform NLU tasks more effectively, some researchers (Wang et al. 2022a, 2023a; Lu et al. 2023; Chen et al. 2022; Zhang et al. 2023) have focused on continuing to train moderate-sized foundation models (approximately 10B parameters, e.g., BLOOM-7B1 (Scao et al. 2023)), which not only improve computational friendliness but also deliver competitive capabilities, in a manner of unifying various tasks. Data consumed in the training procedure can be sourced from either an aggregation of existing close-domain datasets (Wang et al. 2022a, 2023a) or open-domain but noisy datasets generated through approaches such as weak supervision (Lu et al. 2023) and interaction with LLMs (Wang et al. 2023b). The extra training purportedly empowers moderate-sized models to surpass their large-scale counterparts in zero-shot performance across various NLU benchmarks. These tuned models can also provide a stable standard output interface, making evaluation and downstream application convenient.

Our research is in the line of enhancing the NLU ability of LLMs via training but involves a broader range of NLU tasks and incorporates a greater diversity of open-domain

[*] Equal first authorship.
[†] Corresponding authors.
This work was conducted when Tianyu Yu, Chengyue Jiang, Chao Lou, Wei Liu, Jiong Cai, Yangning Li and Yinghui Li were interning at Alibaba DAMO Academy.

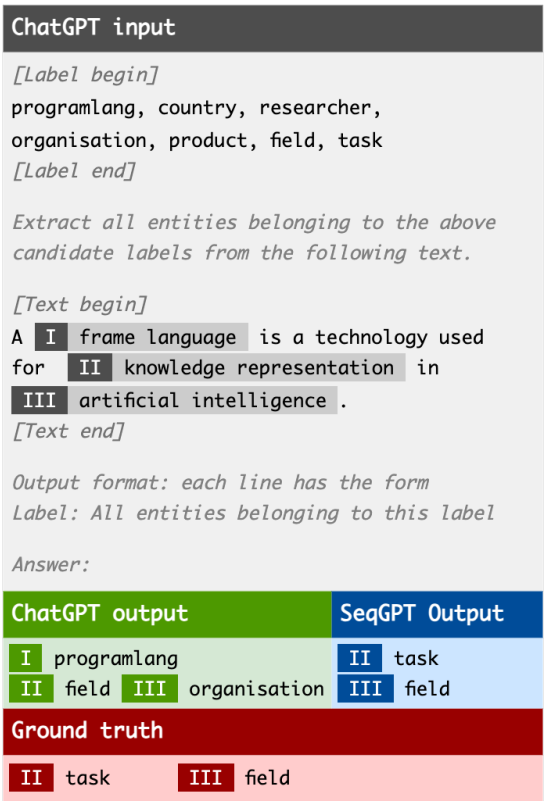

Figure 1: An example of ChatGPT and SeqGPT performing the CrossNER task in the zero-shot setting. ChatGPT mislabeled entities, while SeqGPT succeeded. *Italic gray texts* are the prompt template. SeqGPT uses a different prompt, as shown in Figure 2.

data than previous work. This is motivated by recent instruction tuning studies, which emphasize the advantages of enhancing task diversity rather than simply increasing data volume (Wang et al. 2022b; Iyer et al. 2023). Specifically, we collect and unify 152 datasets across 11 NLU tasks, encompassing not only commonly included information extraction (IE) tasks like NER (Wang et al. 2022a, 2023a), but also tasks overlooked in prior work, such as natural language inference (NLI) and extraction-based machine reading comprehension (MRC). Moreover, to bridge the discrepancy between practical scenarios and existing close-domain NLU data, we generate a large-scale open-domain dataset from various sources. In contrast to earlier studies on automatic NLU data generation, which typically rely on a single domain source (e.g., Wikipedia) and assign labels based on a predefined knowledge base (Lu et al. 2023), we instruct ChatGPT to invent appropriate labels for each sample and identify corresponding answers because ChatGPT is proficient at summarizing and producing annotations at a human level (Brown et al. 2020; Gilardi, Alizadeh, and Kubli 2023; Zhu et al. 2023). The generated dataset contains more than 800 thousand distinct reasonable labels, which is substantially richer than previous datasets but remains high quality upon our manual inspection.

Using the two datasets, we train **Seq**uence understanding enhanced **GPT**, shortly SeqGPT, based on BLOOMZ (Muennighoff et al. 2023), a family of instruction-tuned language models. Our training procedure consists of two stages: initially, pre-training using the diverse, albeit noisy, ChatGPT-generated data and subsequently fine-tuning with the collection of real NLU datasets. This strategy is driven by the intention to first enhance the ability of generalization with diverse data and then refine the model to align with human preferences. Our experiments revealed that SeqGPT consistently surpasses ChatGPT on zero-shot NLU benchmarks by a large margin. The key findings derived from our study can be summarized as follows:

- Scaling up the model size enhances performance.
- Simply scaling up the data size without considering diversity does not consistently yield better performance.
- Increasing task diversity improves performance, although this increase is logarithmic with respect to the number of tasks.
- Larger models are capable of generalizing across languages and tasks.

2 Method

2.1 Unified Approach

In order to solve a novel open-domain task, a language model expects a sequential input encoding both the sentence and necessary knowledge of the task and outputs answers accordingly. To tackle different NLU tasks with a single model and a consistent input-output format, we consider a unified approach that translates them into two atomic tasks:

- **Extraction (EXT):** This task identifies all relevant spans for each query. A query can be a single word, a phrase (as in traditional extraction tasks), or a natural language description (as in machine reading comprehension and instruction following).
- **Classification (CLS):** This task aims to associate the entire input with a suitable subset of the given labels, which permits both multi-class and multi-label classification.

For each atomic task, we design a simple prompt template, which consists of (1) some control tokens indicating different parts of inputs, (2) the specific text to be analyzed, and (3) a list of queries or labels of interest. Regarding the output, the answers are formatted into fixed and easy-to-parse forms depending on the type of atomic tasks. Particularly, for the extraction task, the answer is listed line by line. Each line contains a user-typed query, followed by a list of phrases as its corresponding answer. We do not require the models to provide the positions from which these phrases are extracted, as transformer-based models are not proficient in token counting. For the classification task, the answer is formatted as a single-line list containing answer labels taken from the provided label set.

Typically, most tasks only involve one atomic task. NLI and NER exemplify tasks that rely solely on classification and extraction, respectively. However, some tasks require decomposition into multiple atomic tasks. For example, relation extraction (RE) is performed first to identify spans via

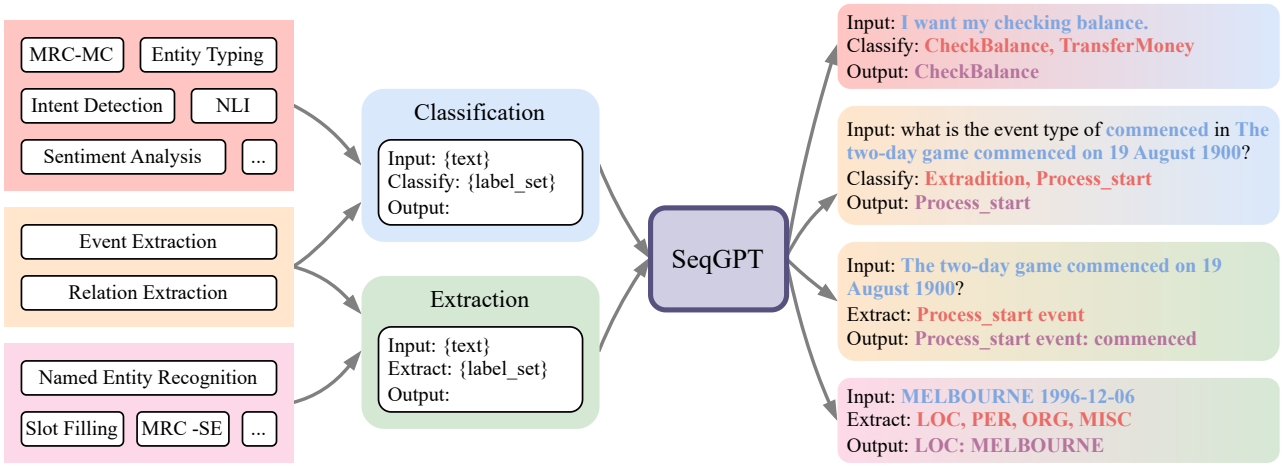

Figure 2: The overview of SeqGPT. Each NLU task is translated into atomic tasks with consistent input-output formats. Black-/blue/red/purple tokens are templates/inputs/query or label lists/outputs.

extraction, followed by classification to discern the relationships between each span pair. Besides, we only make necessary efforts of prompt designing to handle task-specific input formats. Taking NLI as an example, its input contains two sentences (i.e., premise and hypothesis), so we concatenate them with a pre-defined separator. Figure 2 shows a brief illustration, and more details are presented in the appendix.

Contrary to previous studies on instruction tuning that require significant effort to design task descriptions (Wang et al. 2022b, 2023b,a), we inject task-specific information to our models via informative queries or labels. Therefore, the model can be generalized to new tasks and domains without human effort to craft new elaborate task descriptions. While this approach may potentially limit the performance due to the inflexible prior knowledge injection at inference time, our experiments show that, after continuous training on massive NLU tasks, the model learns how to solve NLU tasks and how to generalize, eliminating the need for additional information in the inference time, such that achieves a balance between efficiency and effectiveness.

As prompts are pivotal to achieving high performance, we examine various design possibilities, such as using language-specific or language-agnostic templates. A thorough discussion and experimental comparison will is displayed in the appendix.

2.2 Pre-training Data

Motivated by recent evidence that scaling data diversity benefits models' generalization ability on unseen data (Wang et al. 2022b; Iyer et al. 2023), we construct a large-scale pre-training (PT) dataset with an extremely diverse label set and multiple domains, including Wikipedia, news, and medicine. For covering both atomic tasks, we consider three tasks: classification, entity typing, and NER, whose annotations are created by prompting ChatGPT to invent appropriate labels for each sample and identify corresponding answers in an open-domain setting. Regarding the data quality, we sampled 1% of the generated samples from each domain and assure the average label accuracy for CLS and EXT tasks are higher than 80% and 75% respectively. Finally, the PT dataset encompasses 1,146,271 instances and 817,075 distinct labels. Detailed statistics are shown in Table 1.

Lang.	Task	# inst.	# token	# label
En	CLS	50,172	4,914,471	22,002
	ET	212,734	21,594,057	84,461
	NER	60,094	9,803,353	117,300
Zh	CLS	49,917	7,283,509	32,209
	ET	576,839	170,318,622	143,935
	NER	196,515	46,210,373	417,168
All		1,146,271	260,124,385	817,075[1]

Table 1: Statistics of the pre-training data. # denotes *the number of*. *inst.* denotes *instance*.

Negative Label Generation The PT data generated by ChatGPT cannot be used for training directly because of the lack of negative labels. We adopt a simple strategy: augmenting samples in the PT data with random labels sampled from the set of all labels occurred in the corresponding PT task (i.e., CLS, ET and NER). Due to the large amount of the set (as shown in Table 1), these sampled labels are likely irrelevant to the input sentence, so it is safe to assume the absence of a corresponding answer.

2.3 Fine-tuning Data

To further calibrate models to perform NLU tasks and eliminate effects caused by errors in the PT dataset, we collect massive high-quality NLU datasets from different domains for fine-tuning. As illustrated in Figure 3, our fine-tuning (FT) dataset consists of 110 NLU datasets across two lan-

[1]Labels with the same literal value but from different tasks are considered as different labels.

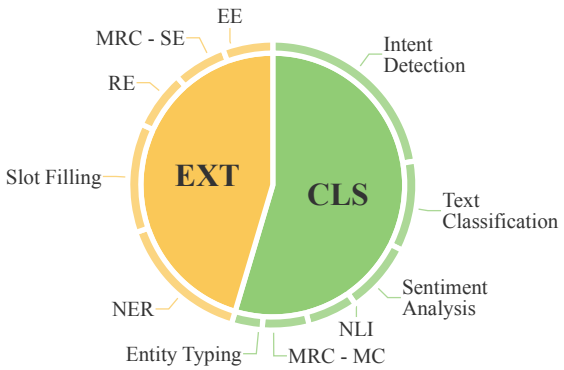

Figure 3: Ratio of each task in the fine-tuning data.

guages, English and Chinese, and ten tasks, including IE tasks, such as NER, RE, and EE and other tasks which can be translated into the two atomic tasks, such as NLI and MRC. Besides a broad coverage of tasks, the data diversity is also guaranteed by their assorted source domains, including medicine, news, and dialogue with AI assistants, and different labels or queries with various granularity. Each task is translated into a combination of atomic tasks, resulting in 139 classification atomic tasks and 94 extraction atomic tasks. We manually select a small portion of the NLU datasets as the held-out set for zero-shot evaluation. We refer readers to the supplementary for the complete list of the included datasets.

Balancing data A large number of datasets are collected in our FT data to ensure diversity, but meanwhile, this introduces data imbalance. Taking two classification datasets as examples, `IFLYTEK` (Xu et al. 2020) and `AG News` (Zhang, Zhao, and LeCun 2015) contains 124 and 31,900 instances per label in average, respectively. In our implementation, we combine collected and sample data uniformly and randomly. The imbalance potentially causes underfitting tasks with abundant samples or oversampling on small datasets. Therefore, we set a quota for each dataset-label pair for balancing data. We use the whole set of instances without up-sampling for those dataset-label pair with fewer instances than the quota.

2.4 Two-stage Training

We train SeqGPT based on BLOOMZ (Muennighoff et al. 2023), an instruction-tuned variant of BLOOM (Scao et al. 2023), with a two-stage training strategy, including pre-training and fine-tuning, as an allusion to the usage of different training data. In our preliminary experiments, this strategy outperforms the alternative: training with a simple mixing of the PT and FT data. Specifically, we use padding to build batches and mask out supervision on the input tokens and train the model with cross-entropy loss. Most hyper-parameters, including optimization steps, learning rates, and batch size, are consistent across all experiments. More training details including hyper-parameters are listed in the appendix to save space.

3 Experiments

3.1 Evaluation

Given the fact that LLMs sometimes generate reasonable but not exactly matched answers, the traditional Micro-F1 metric is not smooth enough for evaluation. To mitigate this and make the evaluation more minor-flaw-tolerant, we propose to combine Micro-F1 and a more smooth ROUGE score as the overall metric. Specifically, we take the average of ROUGE-1, ROUGE-2, and ROUGE-L (Lin 2004) as ROUGE score and take the average of Micro-F1 and ROUGE score as the final score.

To thoroughly evaluate the generalization ability, we evaluate SeqGPT on 233 held-in datasets and 49 held-out datasets. Specifically, the training split of held-in datasets is used during training, no sample from held-out datasets is seen during training, and all tasks involved in held-out datasets are seen during training. For efficiency, we randomly sample 48 records from each evaluation dataset's valid and test split. Besides, in terms of tasks translated to multiple atomic tasks, we simplify the evaluation to report the average scores over atomic tasks. Unless otherwise specified, all scores reported in this section are held-out performance for simplicity.

3.2 Baselines

We compared SeqGPT with the well-known large chat language model ChatGPT (OpenAI 2022) and instruction fine-tuned model series BLOOMZ (Fan et al. 2022).

3.3 Main Results

We compared the held-out performance of the SeqGPT family and baselines in Table 2. Based on the results, we have the following findings:

(1) The smallest SeqGPT-560M surpasses the performance of ChatGPT by a large margin of 27.4, demonstrating the effectiveness of our framework and powerful natural language understanding ability can be learned by a compact small model. On the other hand, the overall score of ChatGPT might be hindered by the metric we adopted since the output format generated by ChatGPT is not always aligned with our evaluation data format. Besides, ChatGPT sometimes can not comprehend prompts, resulting in irrelevant responses. We refer readers to Section 3.7 for a more detailed analysis of comparing ChatGPT with SeqGPT.

(2) The average score can be further improved to 65.5 by using a larger 7B1 backbone. This improvement can be attributed to better complex reasoning ability and more diverse world knowledge that comes with larger models.

(3) The weakly supervised ultra-fine-grained pre-training data are helpful, especially for smaller models like SeqGPT 560M. Without using the pre-training data, the average performance of SeqGPT 560M drops from 57.2 to 53.9. Specifically, the scores of entity typing and slot filling, which require a diverse range of understanding of entities, drops significantly for SeqGPT of all sizes.

Model	Size	CLS	EE	ID	MRC	NER	NLI	RE	SF	SA	ET	ALL
ChatGPT	-	58.0	34.8	62.3	19.9	11.1	33.5	31.4	30.6	65.6	27.9	38.1
BLOOMZ	560M	5.3	1.6	3.6	4.4	0.0	5.8	0.7	0.0	11.30	3.3	3.6
	1B7	5.6	2.4	0.9	3.8	0.0	10.1	4.3	0.0	16.0	3.5	3.7
	3B	6.8	3.9	1.8	4.4	0.0	4.4	3.3	0.0	12.5	3.6	4.7
	7B1	10.3	6.2	2.4	6.4	0.0	14.0	11.2	0.2	24.6	4.2	6.2
SeqGPT w/o pre-training	560M	53.7	48.0	64.1	39.1	48.9	48.7	40.5	66.1	71.2	32.8	53.9
	1B7	62.5	55.1	78.0	45.1	52.0	52.9	50.4	65.4	**78.5**	34.2	60.1
	3B	65.9	59.7	79.9	45.4	53.8	57.9	51.6	70.1	76.0	37.4	62.2
	7B1	**72.7**	**63.4**	**83.3**	49.2	55.5	60.4	**57.4**	71.7	73.5	43.1	65.4
SeqGPT	560M	57.3	56.8	72.9	38.8	50.9	51.4	43.9	70.0	71.7	38.8	57.2
	1B7	67.9	57.2	80.9	43.8	52.7	57.5	56.7	70.1	77.2	48.1	62.8
	3B	68.5	60.9	77.2	48.8	54.8	**62.5**	54.3	**75.1**	73.1	48.9	64.0
	7B1	70.9	63.1	80.9	**51.0**	**56.1**	58.9	56.0	72.1	74.3	**54.1**	**65.5**

Table 2: Performance on held-out evaluation datasets. CLS: text classification. EE: event extraction. ID: intent detection; MRC: machine reading comprehension. NER: named-entity recognition. NLI: natural language inference. RE: relation extraction. SF: slot filling. SA: sentiment analysis. ET: entity typing. ALL: average performance on all tasks.

(4) Though effective, the performance gains achieved by utilizing pre-training data shrinks with larger models. We argue that this is because the ultra-fine-grained knowledge in our pre-training data can also be learned directly during the pre-training stage of LLMs, and such knowledge is better learned with increasing model size of pre-trained LLMs. On the other hand, the naive BLOOMZ 7B1 lags far behind even the smallest SeqGPT 560M. We find the output generated by BLOOMZ 7B1 can hardly be consistent with the instruction, indicating complex prompt engineering or few-shot examples might be required to leverage such general instruction following model to solve open-domain NLU tasks.

3.4 Scaling Analysis

We extensively study the performance of models with respect to the scaling of model sizes, number of samples per task, and number of distinct tasks and discover all these factors are crucial for building an open-domain sequence understanding model.

Model Size We trained a series of models in different sizes based on the BLOOMZ family (Fan et al. 2022) from 560M to 7B1 to explore the scaling effect of model sizes. Results in Figure 4 show both the held-in and the held-out performance increase with a larger backbone that complies with the results found in Chowdhery et al. (2022). Furthermore, the large gap between the held-in and held-out performance reveals the difficulty of open-domain NLU, indicating that there is still great space for SeqGPT to improve the generalization ability. We find the improvement in held-in evaluation is fewer compared with the held-out evaluation. We believe the held-out score can better reflect the performance in real applications.

Number of Training Datasets Besides the model size, the number of training datasets is also the major factor to impact the resulting performance, so we also conduct experiments to explore this effect. Results in Figure 5 indicate that

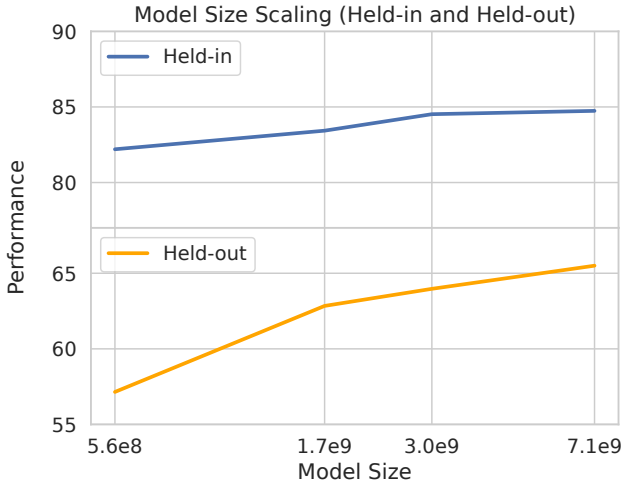

Figure 4: Held-in and held-out evaluation results of SeqGPT in different sizes.

the performance of our SeqGPT models increases in a logarithmic manner with more datasets used for training. Based on such observation, we believe that adding more training datasets is an efficient and straightforward approach to improve the performance further since our held-in corpora are still small compared to opulent real application scenarios.

3.5 Cross-language Generalization

We use a great amount of training data from both English and Chinese. To explore the effect of data from each language and the cross-language generalization ability of SeqGPT, we conduct extensive experiments, and the main results are shown in Table 3. We find that the models trained with a single language (English/Chinese) can generalize to tasks in the other language (Chinese/English) and achieve

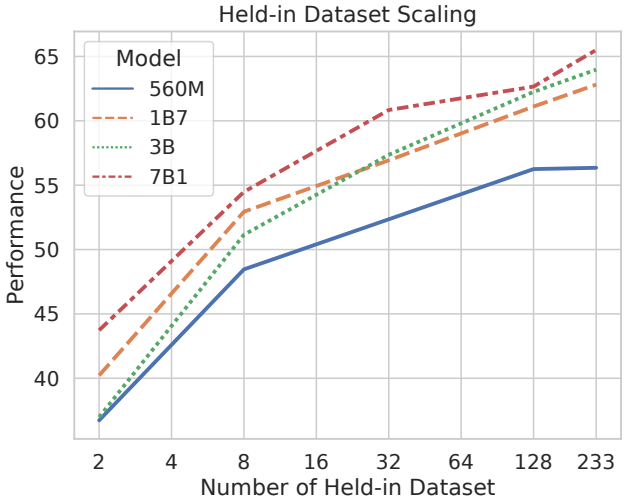

Figure 5: Held-out performance of SeqGPT in different sizes scaling with respect to the number of training datasets in the held-in set.

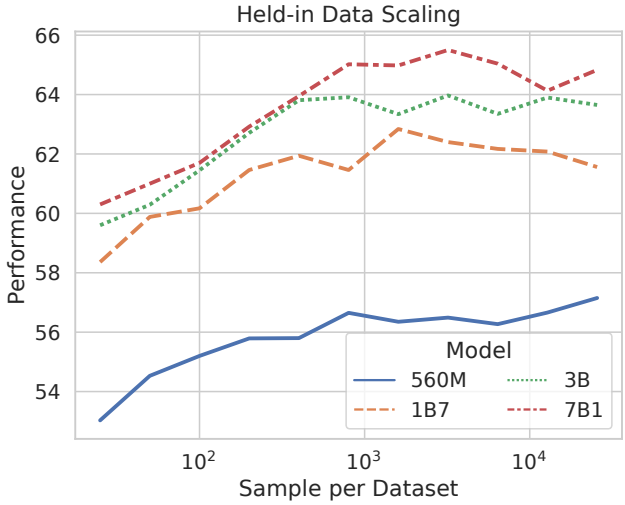

Figure 6: Held-out performance of SeqGPT scaling with respect to the number of samples per dataset.

Training Languages	EN Score	ZH Score
English	57.59	51.98
Chinese	52.66	64.57
Chinese + English	**58.83**	**65.23**

Table 3: Performance of SeqGPT-1B7 trained with different settings of training languages.

reasonable performance. Comparing the model trained with data in English and in both languages, we find the scores on both English tasks and Chinese tasks can be improved, showing there are skills shared between languages that can be learned through a multilingual training stage.

3.6 Cross-task Generalization

Though sharing mostly the same prompts in our framework, the skills needed to solve different tasks is diverse. To analyze how SeqGPT works on tasks not seen during training and how the training task affects the performance of different test tasks, we train a series of models with only one task, and results are shown in Figure 7. Based on the results we find models achieve the best evaluation performance when the evaluation task is the same as the training task except for the NLI task. For NLI performance, we find the model trained on the NLI task even achieves the worst performance. We argue this is because the way to classify sentence pairs differs across NLI datasets. As a result, models trained on only NLI datasets can hardly transfer the classification boundaries learned from the held-in datasets to held-out datasets. Models trained on EE, MRC, and RE can generalize well to all test tasks, demonstrating the diverse knowledge required to solve these tasks are also crucial for other tasks and can serve as a great training resource for models targeting general domain NLU.

3.7 Human Evaluation

For a more comprehensive analysis, we perform a human evaluation on the held-out datasets. The evaluation recruits ten well-educated annotators and presents them with answers generated by ChatGPT and SeqGPT-7B1. Annotators are required to decide which model gives the better answer or two models are tied with each other. Results are shown in Figure 8. From the results, we can find that SeqGPT-7B1 achieves higher performance on seven out of ten NLU tasks, demonstrating the effectiveness of training the model with a wide range of NLU tasks incorporating a great diversity of open-domain data. Also, we found the output of SeqGPT-7B1 is much more concise than the output of ChatGPT, making the interpretation easier and consequently reducing the engineering complexity to use the model. However, the results also indicate that medium-size models like SeqGPT-7B1 still lack the complex reasoning abilities to solve complicated tasks such as NER and SF.

4 Related Work

4.1 Large Language Models

Autoregressive language models have rapidly scaled up, reaching billions of parameters and trillions of training tokens. This has resulted in many emergent abilities such as few-shot learning, in-context learning, and reasoning (Bubeck et al. 2023; Wei et al. 2022b). Examples include GPT-3 (Brown et al. 2020), Chinchilla (Hoffmann et al. 2022), Llama (Touvron et al. 2023a,b) and BLOOM (Scao et al. 2023). LLMs can be prompted to perform downstream tasks without training, such as ChatIE for IE tasks (Wei et al. 2023), PromptNER for NER tasks (Ashok and Lipton 2023) and Liu et al. (2023) for text-to-SQL tasks. We refer the readers to (Zheng et al. 2023; Li et al. 2023a) and references therein for more details.

In this study, we adopt BLOOMZ (Muennighoff et al. 2023), a BLOOM-based instruction-tuned model, as the

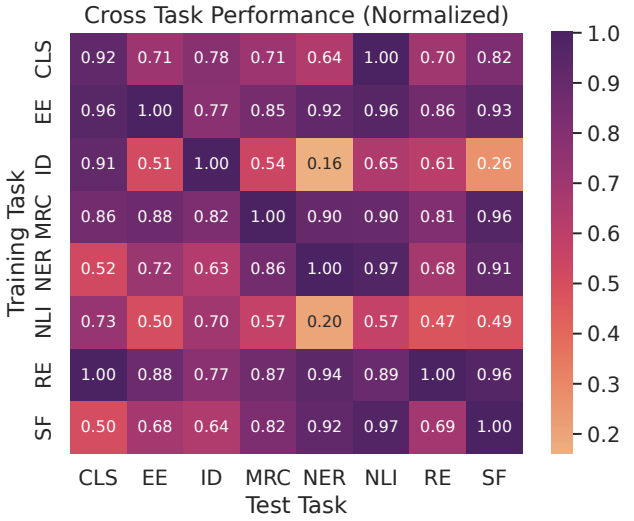

Figure 7: Cross task generalization experiment results. Scores are normalized column-wise based on the max score of each column.

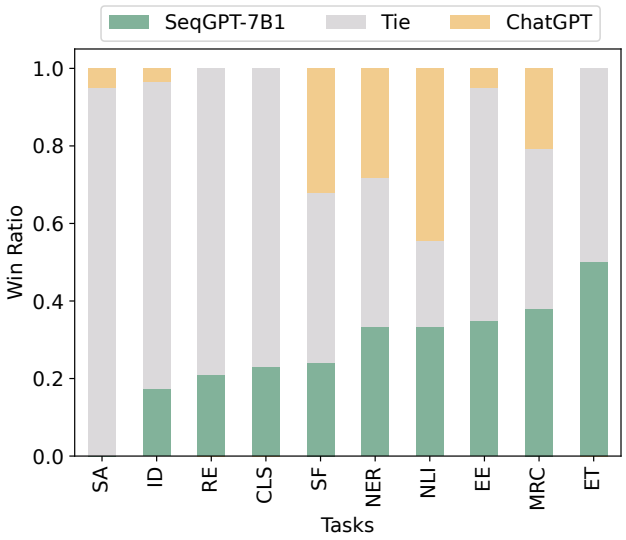

Figure 8: Human evaluation on held-out datasets.

backbone due to its exceptional multilingual performance among publicly available models and superior generalization capabilities compared to BLOOM.

4.2 Instruction Tuning

Instruction tuning (Wei et al. 2022a; Wang et al. 2022b; Sanh et al. 2022; Li et al. 2023e) is a novel finetuning paradigm that trains language models on numbers of tasks described using natural language instructions. It has shown potential benefits in aligning better with human preferences, yielding more truthful, useful, and less harmful output (Ouyang et al. 2022; Lou, Zhang, and Yin 2023). Furthermore, it has demonstrated enhanced task-specific performance (Longpre et al. 2023; Jang et al. 2023; Ivison et al. 2023) even tuning only on a single task (Lee et al. 2023; Gupta et al. 2023; Chen et al. 2023), as well as generalization capabilities for unseen tasks (Wang et al. 2022b, 2023b). Most instruction-tuning methods leverage datasets covering some NLU tasks but with poor coverage of tasks and domains. For a specialized model, Wang et al. (2023a) train InstructUIE on wide IE tasks with various instructions and Parmar et al. (2022) build a biomedical LLM with a collection of biomedical datasets across multiple tasks with human-crafted instructions.

4.3 Unified Models for NLU

Diverse NLU tasks emphasize different aspects of languages. Multitask learning has emerged as a prevalent topic, taking advantage of jointly modeling selected subsets of NLU tasks, such as enabling the use of more training data or modeling similarities between tasks (Thrun 1995; Caruana 1997; Miller et al. 2000; Sutton, McCallum, and Rohanimanesh 2007; Liu, Qiu, and Huang 2016; Liu et al. 2019; Lu et al. 2022a, among others). When incorporating more tasks, sequence generation models become compelling options because free texts may be the most straightforward way to encode all outputs of various NLU tasks. UIE (Lu et al. 2022b) unify the inputs of IE tasks through a schema-based prompt mechanism and the outputs through the novel structural extraction language. Consequently, given suitable prompts, it can perform novel NLU tasks using the common semantic understanding ability learned. Subsequently, InstructUIE (Wang et al. 2023a) extends UIE by instruction tuning a stronger backbone model (e.g., Flan-T5 11B), showing strong zero-shot performance. USM (Lou et al. 2023) is another unified IE model based on a link prediction mechanism named semantic matching.

5 Conclusions

In this study, we introduce SeqGPT, a unified model devised to handle various NLU tasks by translating different NLU tasks into two common atomic tasks. In this way, SeqGPT offers a consistent input-output format, enabling it to solve unseen tasks by prompting arbitrarily varied label sets without tedious prompt engineering. To achieve strong generalization ability, we train the model using novel ultra fine-grained synthetic data and a massive collection of NLU datasets on various domains. The training is further enhanced with effective data balance and randomly sampled negative labels. Both automatic benchmarks and human evaluation on unseen tasks show that SeqGPT achieves consistent improvements over ChatGPT. In addition, we conduct comprehensive experiments to investigate behaviors of scaling, revealing a logarithmic correlation between the quantity of training tasks and model performance. We have also evaluated SeqGPT's ability to generalize across various tasks and languages. Nevertheless, our findings raise new questions. Why does the PT data fail to enhance SeqGPT-7B1, while an increase in FT data does? How to generate more high-quality NLU data to fill the data hunger of SeqGPT? We hope future research on these questions to further improve open-domain NLU models.

Acknowledgments

This research is supported by National Natural Science Foundation of China (Grant No.62276154), Research Center for Computer Network (Shenzhen) Ministry of Education, the Natural Science Foundation of Guangdong Province (Grant No. 2023A1515012914), Basic Research Fund of Shenzhen City (Grant No. JCYJ20210324120012033 and JSGG20210802154402007), the Major Key Project of PCL for Experiments and Applications (PCL2021A06), and Overseas Cooperation Research Fund of Tsinghua Shenzhen International Graduate School (HW2021008). This work is partially supported by the Shenzhen Science and Technology Program (WDZC20231128091437002).

References

Ashok, D.; and Lipton, Z. C. 2023. PromptNER: Prompting For Named Entity Recognition.

Brown, T. B.; Mann, B.; Ryder, N.; et al. 2020. Language Models are Few-Shot Learners.

Bubeck, S.; Chandrasekaran, V.; Eldan, R.; Gehrke, J.; Horvitz, E.; Kamar, E.; Lee, P.; Lee, Y. T.; Li, Y.; Lundberg, S.; Nori, H.; Palangi, H.; Ribeiro, M. T.; and Zhang, Y. 2023. Sparks of Artificial General Intelligence: Early experiments with GPT-4.

Caruana, R. 1997. Multitask Learning. *Machine Learning*.

Chase, H. 2022. LangChain. Original-date: 2022-10-17T02:58:36Z.

Chen, H.; Zhang, Y.; Zhang, Q.; Yang, H.; Hu, X.; Ma, X.; Yanggong, Y.; and Zhao, J. 2023. Maybe Only 0.5

Chen, X.; Zhang, N.; Xie, X.; Deng, S.; Yao, Y.; Tan, C.; Huang, F.; Si, L.; and Chen, H. 2022. KnowPrompt: Knowledge-Aware Prompt-Tuning with Synergistic Optimization for Relation Extraction. In *Proceedings of the ACM Web Conference 2022*, WWW '22.

Chowdhery, A.; Narang, S.; Devlin, J.; et al. 2022. PaLM: Scaling Language Modeling with Pathways.

Chung, H. W.; Hou, L.; Longpre, S.; et al. 2022. Scaling Instruction-Finetuned Language Models.

Fan, A.; Ilic, S.; Wolf, T.; and Gallé, M., eds. 2022. *Proceedings of BigScience Episode #5 – Workshop on Challenges & Perspectives in Creating Large Language Models*. virtual+Dublin: Association for Computational Linguistics.

Gilardi, F.; Alizadeh, M.; and Kubli, M. 2023. ChatGPT Outperforms Crowd-Workers for Text-Annotation Tasks.

Gravitas, S. 2023. AutoGPT.

Gupta, H.; Sawant, S. A.; Mishra, S.; Nakamura, M.; Mitra, A.; Mashetty, S.; and Baral, C. 2023. Instruction Tuned Models are Quick Learners.

Hoffmann, J.; Borgeaud, S.; Mensch, A.; Buchatskaya, E.; Cai, T.; Rutherford, E.; de Las Casas, D.; Hendricks, L. A.; Welbl, J.; Clark, A.; Hennigan, T.; Noland, E.; Millican, K.; van den Driessche, G.; Damoc, B.; Guy, A.; Osindero, S.; Simonyan, K.; Elsen, E.; Rae, J. W.; Vinyals, O.; and Sifre, L. 2022. Training Compute-Optimal Large Language Models.

Hu, E. J.; yelong shen; Wallis, P.; Allen-Zhu, Z.; Li, Y.; Wang, S.; Wang, L.; and Chen, W. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations*.

Ivison, H.; Smith, N. A.; Hajishirzi, H.; and Dasigi, P. 2023. Data-Efficient Finetuning Using Cross-Task Nearest Neighbors. In *Findings of ACL*.

Iyer, S.; Lin, X. V.; Pasunuru, R.; Mihaylov, T.; Simig, D.; Yu, P.; Shuster, K.; Wang, T.; Liu, Q.; Koura, P. S.; Li, X.; O'Horo, B.; Pereyra, G.; Wang, J.; Dewan, C.; Celikyilmaz, A.; Zettlemoyer, L.; and Stoyanov, V. 2023. OPT-IML: Scaling Language Model Instruction Meta Learning through the Lens of Generalization.

Jang, J.; Kim, S.; Ye, S.; Kim, D.; Logeswaran, L.; Lee, M.; Lee, K.; and Seo, M. 2023. Exploring the Benefits of Training Expert Language Models over Instruction Tuning.

Kaplan, J.; McCandlish, S.; Henighan, T.; Brown, T. B.; Chess, B.; Child, R.; Gray, S.; Radford, A.; Wu, J.; and Amodei, D. 2020. Scaling Laws for Neural Language Models.

Lee, Y.-S.; Astudillo, R. F.; Florian, R.; Naseem, T.; and Roukos, S. 2023. AMR Parsing with Instruction Fine-tuned Pre-trained Language Models.

Li, X.; Zhang, T.; Dubois, Y.; Taori, R.; Gulrajani, I.; Guestrin, C.; Liang, P.; and Hashimoto, T. B. 2023a. AlpacaEval: An Automatic Evaluator of Instruction-following Models.

Li, Y.; Chen, J.; Li, Y.; Xiang, Y.; Chen, X.; and Zheng, H.-T. 2023b. Vision, Deduction and Alignment: An Empirical Study on Multi-Modal Knowledge Graph Alignment. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1–5. IEEE.

Li, Y.; Huang, H.; Ma, S.; Jiang, Y.; Li, Y.; Zhou, F.; Zheng, H.; and Zhou, Q. 2023c. On the (In)Effectiveness of Large Language Models for Chinese Text Correction. *CoRR*.

Li, Y.; Li, Y.; Chen, X.; Zheng, H.-T.; and Shen, Y. 2023d. Active relation discovery: Towards general and label-aware open relation extraction. *Knowledge-Based Systems*, 282: 111094.

Li, Y.; Ma, S.; Wang, X.; Huang, S.; Jiang, C.; Zheng, H.-T.; Xie, P.; Huang, F.; and Jiang, Y. 2023e. EcomGPT: Instruction-tuning Large Language Model with Chain-of-Task Tasks for E-commerce. *arXiv preprint arXiv:2308.06966*.

Lin, C.-Y. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*, 74–81. Barcelona, Spain: Association for Computational Linguistics.

Liu, A.; Hu, X.; Wen, L.; and Yu, P. S. 2023. A comprehensive evaluation of ChatGPT's zero-shot Text-to-SQL capability.

Liu, H.; Tam, D.; Mohammed, M.; Mohta, J.; Huang, T.; Bansal, M.; and Raffel, C. 2022a. Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning. In *NeurIPS*.

Liu, P.; Qiu, X.; and Huang, X. 2016. Recurrent Neural Network for Text Classification with Multi-Task Learning. *ArXiv*.

Liu, X.; He, P.; Chen, W.; and Gao, J. 2019. Multi-Task Deep Neural Networks for Natural Language Understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 4487–4496. Florence, Italy: Association for Computational Linguistics.

Liu, X.; Ji, K.; Fu, Y.; Tam, W.; Du, Z.; Yang, Z.; and Tang, J. 2022b. P-Tuning: Prompt Tuning Can Be Comparable to Fine-tuning Across Scales and Tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 61–68. Dublin, Ireland: Association for Computational Linguistics.

Longpre, S.; Hou, L.; Vu, T.; Webson, A.; Chung, H. W.; Tay, Y.; Zhou, D.; Le, Q. V.; Zoph, B.; Wei, J.; and Roberts, A. 2023. The Flan Collection: Designing Data and Methods for Effective Instruction Tuning.

Lou, J.; Lu, Y.; Dai, D.; Jia, W.; Lin, H.; Han, X.; Sun, L.; and Wu, H. 2023. Universal Information Extraction as Unified Semantic Matching. *AAAI*.

Lou, R.; Zhang, K.; and Yin, W. 2023. Is Prompt All You Need? No. A Comprehensive and Broader View of Instruction Learning.

Lu, J.; Yang, P.; Gan, R.; Yang, J.; and Zhang, J. 2022a. Unified BERT for Few-shot Natural Language Understanding.

Lu, K.; Pan, X.; Song, K.; Zhang, H.; Yu, D.; and Chen, J. 2023. PIVOINE: Instruction Tuning for Open-world Information Extraction.

Lu, Y.; Liu, Q.; Dai, D.; Xiao, X.; Lin, H.; Han, X.; Sun, L.; and Wu, H. 2022b. Unified Structure Generation for Universal Information Extraction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 5755–5772. Dublin, Ireland: Association for Computational Linguistics.

Miller, S.; Fox, H.; Ramshaw, L.; and Weischedel, R. 2000. A Novel Use of Statistical Parsing to Extract Information from Text. In *1st Meeting of the North American Chapter of the Association for Computational Linguistics*.

Muennighoff, N.; Wang, T.; Sutawika, L.; Roberts, A.; Biderman, S.; Scao, T. L.; Bari, M. S.; Shen, S.; Yong, Z.-X.; Schoelkopf, H.; Tang, X.; Radev, D.; Aji, A. F.; Almubarak, K.; Albanie, S.; Alyafeai, Z.; Webson, A.; Raff, E.; and Raffel, C. 2023. Crosslingual Generalization through Multitask Finetuning.

OpenAI, T. 2022. Chatgpt: Optimizing language models for dialogue. *OpenAI*.

Ouyang, L.; Wu, J.; Jiang, X.; Almeida, D.; Wainwright, C. L.; Mishkin, P.; Zhang, C.; Agarwal, S.; Slama, K.; Ray, A.; Schulman, J.; Hilton, J.; Kelton, F.; Miller, L.; Simens, M.; Askell, A.; Welinder, P.; Christiano, P.; Leike, J.; and Lowe, R. 2022. Training language models to follow instructions with human feedback.

Parmar, M.; Mishra, S.; Purohit, M.; Luo, M.; Mohammad, M.; and Baral, C. 2022. In-BoXBART: Get Instructions into Biomedical Multi-Task Learning. In *Findings of the Association for Computational Linguistics: NAACL 2022*, 112–128. Seattle, United States: Association for Computational Linguistics.

Qin, C.; Zhang, A.; Zhang, Z.; Chen, J.; Yasunaga, M.; and Yang, D. 2023. Is ChatGPT a General-Purpose Natural Language Processing Task Solver?

Sanh, V.; Webson, A.; Raffel, C.; et al. 2022. Multitask Prompted Training Enables Zero-Shot Task Generalization. In *International Conference on Learning Representations*.

Scao, T. L.; Fan, A.; Akiki, C.; et al. 2023. BLOOM: A 176B-Parameter Open-Access Multilingual Language Model.

Sutton, C.; McCallum, A.; and Rohanimanesh, K. 2007. Dynamic Conditional Random Fields: Factorized Probabilistic Models for Labeling and Segmenting Sequence Data. *Journal of Machine Learning Research*.

Thrun, S. 1995. Is Learning The n-th Thing Any Easier Than Learning The First? In Touretzky, D.; Mozer, M.; and Hasselmo, M., eds., *NeurIPS*.

Touvron, H.; Lavril, T.; Izacard, G.; Martinet, X.; Lachaux, M.-A.; Lacroix, T.; Rozière, B.; Goyal, N.; Hambro, E.; Azhar, F.; Rodriguez, A.; Joulin, A.; Grave, E.; and Lample, G. 2023a. LLaMA: Open and Efficient Foundation Language Models.

Touvron, H.; Martin, L.; Stone, K.; et al. 2023b. Llama 2: Open Foundation and Fine-Tuned Chat Models.

Wadhwa, S.; Amir, S.; and Wallace, B. 2023. Revisiting Relation Extraction in the era of Large Language Models. In *ACL*.

Wang, C.; Liu, X.; Chen, Z.; Hong, H.; Tang, J.; and Song, D. 2022a. DeepStruct: Pretraining of Language Models for Structure Prediction. In *Findings of the Association for Computational Linguistics: ACL 2022*, 803–823. Dublin, Ireland: Association for Computational Linguistics.

Wang, X.; Zhou, W.; Zu, C.; Xia, H.; Chen, T.; Zhang, Y.; Zheng, R.; Ye, J.; Zhang, Q.; Gui, T.; Kang, J.; Yang, J.; Li, S.; and Du, C. 2023a. InstructUIE: Multi-task Instruction Tuning for Unified Information Extraction.

Wang, Y.; Kordi, Y.; Mishra, S.; Liu, A.; Smith, N. A.; Khashabi, D.; and Hajishirzi, H. 2023b. Self-Instruct: Aligning Language Models with Self-Generated Instructions. In *ACL*.

Wang, Y.; Mishra, S.; Alipoormolabashi, P.; et al. 2022b. Super-NaturalInstructions: Generalization via Declarative Instructions on 1600+ NLP Tasks. In *EMNLP*.

Wei, J.; Bosma, M.; Zhao, V. Y.; Guu, K.; Yu, A. W.; Lester, B.; Du, N.; Dai, A. M.; and Le, Q. V. 2022a. Finetuned Language Models Are Zero-Shot Learners.

Wei, J.; Tay, Y.; Bommasani, R.; Raffel, C.; Zoph, B.; Borgeaud, S.; Yogatama, D.; Bosma, M.; Zhou, D.; Metzler, D.; Chi, E. H.; Hashimoto, T.; Vinyals, O.; Liang, P.; Dean, J.; and Fedus, W. 2022b. Emergent Abilities of Large Language Models.

Wei, X.; Cui, X.; Cheng, N.; Wang, X.; Zhang, X.; Huang, S.; Xie, P.; Xu, J.; Chen, Y.; Zhang, M.; Jiang, Y.; and Han,

W. 2023. Zero-Shot Information Extraction via Chatting with ChatGPT.

Xu, L.; Hu, H.; Zhang, X.; Li, L.; Cao, C.; Li, Y.; Xu, Y.; Sun, K.; Yu, D.; Yu, C.; Tian, Y.; Dong, Q.; Liu, W.; Shi, B.; Cui, Y.; Li, J.; Zeng, J.; Wang, R.; Xie, W.; Li, Y.; Patterson, Y.; Tian, Z.; Zhang, Y.; Zhou, H.; Liu, S.; Zhao, Z.; Zhao, Q.; Yue, C.; Zhang, X.; Yang, Z.; Richardson, K.; and Lan, Z. 2020. CLUE: A Chinese Language Understanding Evaluation Benchmark. In *Proceedings of the 28th International Conference on Computational Linguistics*, 4762–4772. Barcelona, Spain (Online): International Committee on Computational Linguistics.

Zhang, N.; Zhang, J.; Wang, X.; et al. 2023. DeepKE-LLM: A Large Language Model Based Knowledge Extraction Toolkit. *GitHub repository*.

Zhang, X.; Zhao, J.; and LeCun, Y. 2015. Character-level Convolutional Networks for Text Classification. In *NeurIPS*.

Zheng, L.; Chiang, W.-L.; Sheng, Y.; Zhuang, S.; Wu, Z.; Zhuang, Y.; Lin, Z.; Li, Z.; Li, D.; Xing, E. P.; Zhang, H.; Gonzalez, J. E.; and Stoica, I. 2023. Judging LLM-as-a-judge with MT-Bench and Chatbot Arena.

Zhu, Y.; Zhang, P.; Haq, E.-U.; Hui, P.; and Tyson, G. 2023. Can ChatGPT Reproduce Human-Generated Labels? A Study of Social Computing Tasks.