Benchmarking Complex Instruction-Following with Multiple Constraints Composition

Bosi Wen[1], Pei Ke[3], Xiaotao Gu[2], Lindong Wu[2], Hao Huang[2], Jinfeng Zhou[1], Wenchuang Li[4], Binxin Hu[5], Wendy Gao[2], Jiaxin Xu[1], Yiming Liu[1], Jie Tang[1], Hongning Wang[1], Minlie Huang[1]

[1]Tsinghua University [2]Zhipu AI [3]The University of Electronic Science and Technology of China

[4]The China University of Geosciences

[5]Central China Normal University

Motivation

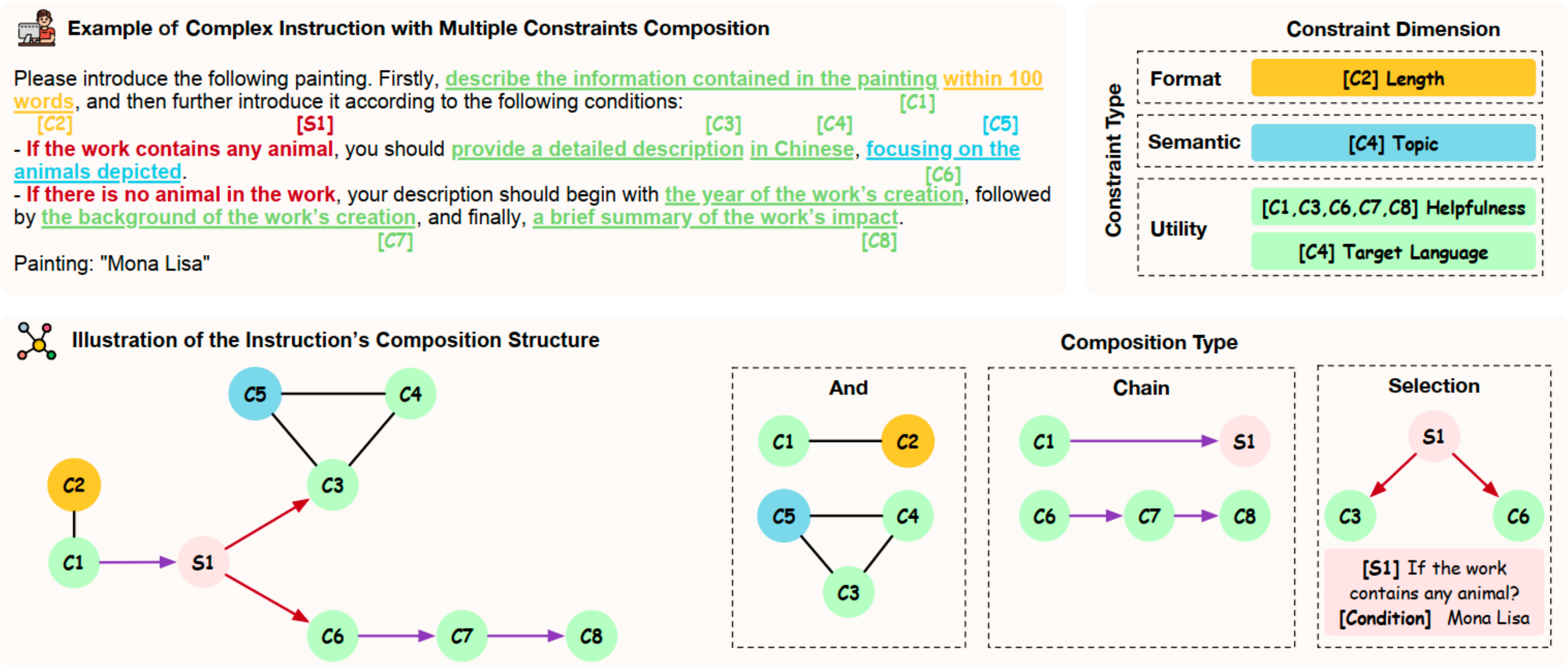
- LLMs have been increasingly applied to deal with complex human instructions in real-world scenarios. Evaluating the complex instruction following capability of LLMs is an important problem.

- Previous benchmarks focus on measuring whether the generated text of LLMs can meet every constraint in the input instruction. However, they neglect to model the composition of constraints, resulting in:
  - Incomprehensive coverage: They are limited to simple composition types such as *And,* which represents coordination between different constraints, failing to cover other composition types of constraints.
  - Bias in evaluation: They assign the same weight to different constraints during score aggregation, ignoring their dependencies and structures.

- Therefore, we propose ComplexBench, a novel benchmark to comprehensively evaluate the ability of LLMs to follow complex instructions.

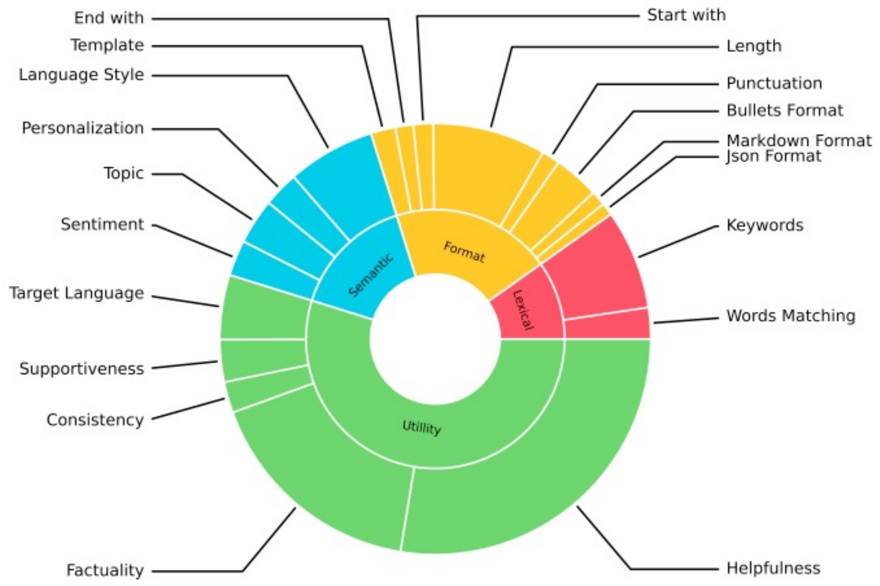
Motivation

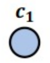
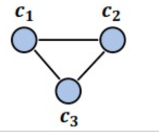
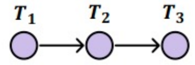
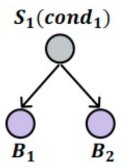
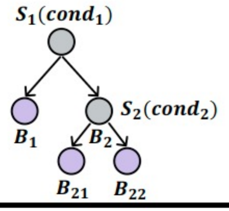
- An example of instruction with multiple constraints composition



Framework

- ComplexBench proposes a hierarchical taxonomy to define constraints and their composition type, including 4 constraint types, 19 constraint dimensions, and 4 composition types.



Composition Type	Description	Example	Illustration
Single	The output is required to satisfy a single constraint.	Please summarize the following news.	
And	The output is required to satisfy multiple constraints simultaneously.	Please summarize the following news. The summary should be output in bullet points, and within 100 words.	
Chain	The output is required to complete multiple tasks sequentially, each of which needs to satisfy its own constraints.	Please introduce "Mona Lisa" briefly. Firstly, introduce the year of creation, then describe the background of the work's creation, and finally, summarize the impact of the work.	
Selection	The output is required to select different branches according to certain conditions, fulfilling the constraints of the corresponding branch.	Please introduce the following painting. - If the work contains any animal, the description should be in English - Otherwise, the description should be in Chinese Painting: "Mona Lisa"	
Nested Structure	The above composition types are recursively nested to form more complex structures.	Analy the sentiment of above user comment and complete the following tasks: 1. If it's positive: - Identify the products within the comments … 2. If it's negative, analyze the reasons for it: - If the reason is not about products itself, … - Otherwise, …	

Data Construction

- The construction pipeline of ComplexBench

Reference Instruction Collection → Task Allocation → Data Annotation and Validation → Selection Branch Expansion

- Overall, ComplexBench consists of 1150 meticulously curated instructions, significantly larger than the previous instruction-following benchmark

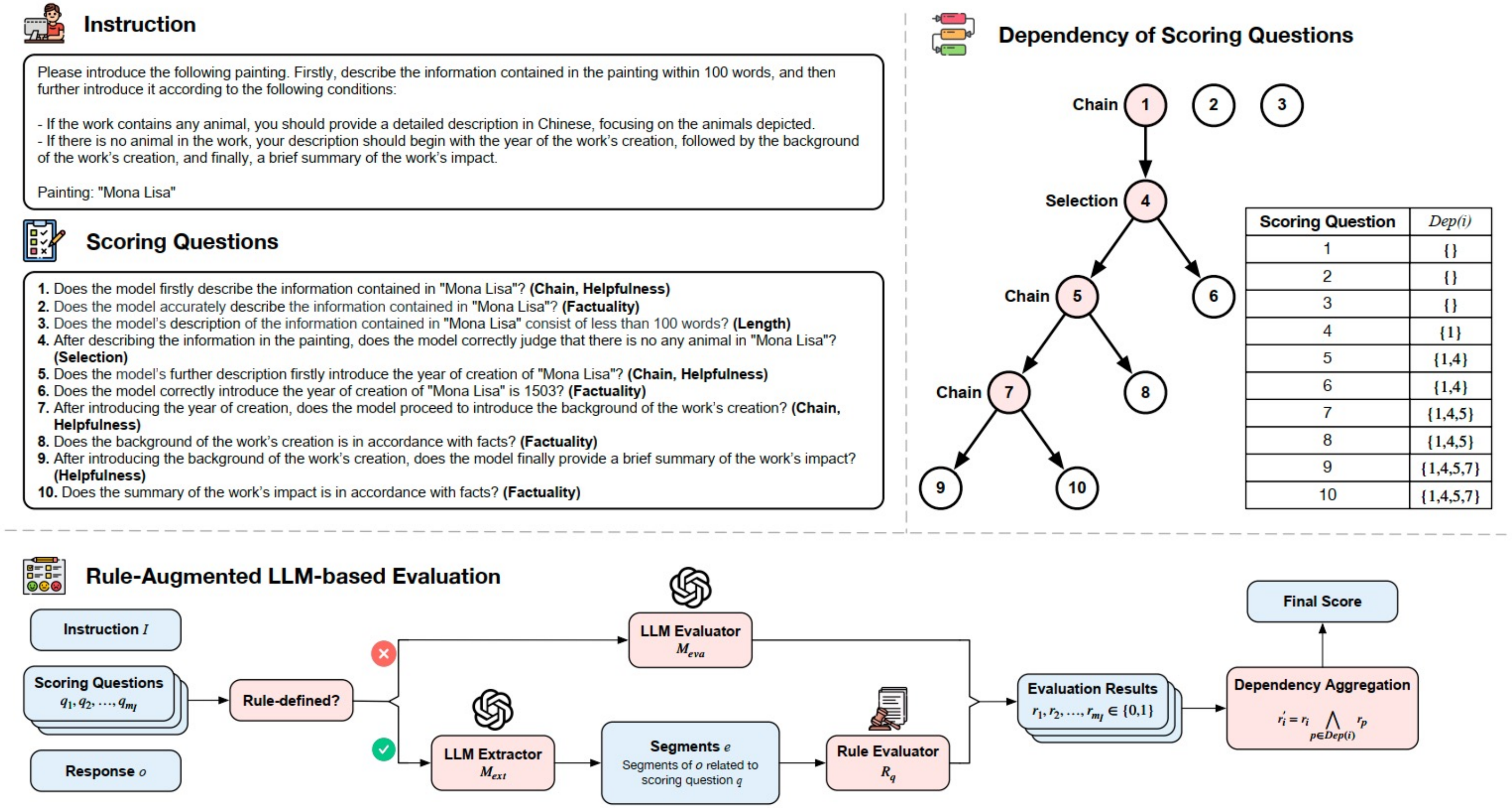
- We categorize ComplexBench based on the included composition types and their nesting depth within instructions.

Category	Nesting Depth	**#Inst.**	**#Len.**	**#Ques.**	**#Con.**
And	1	475	279.39	4.09	4.14
Chain	1	70	352.11	4.83	4.94
	2	170	486.84	6.24	6.32
Selection	1	80	753.15	2.91	2.06
	2	224	664.13	4.40	3.09
	≥ 3	46	1409.93	5.76	3.78
Selection & Chain	2	30	440.37	4.37	3.63
	≥ 3	55	398.82	6.18	5.27
Overall	-	1150	477.51	4.61	4.19

Table 2: Statistics of ComplexBench including the number of instructions (**#Inst.**), the average number of characters (**#Len.**), scoring questions (**#Ques.**), and constraints (**#Con.**) per instruction.

Evaluation Protocol

- Design a yes / no question to verify each constraint and composition type respectively
- RAL: Equip LLM evaluators with rules to answer scoring questions in both rule-defined and open-ended areas
- Model the dependencies of scoring questions based on the composition types



Experiment: Main Results

- By evaluating 15 closed-source and open-source popular LLMs on ComplexBench, we highlight the weaknesses of LLMs in following complex instructions and point toward potential avenues for future work

Category	And	Chain			Selection				Selection & Chain			All
Nesting Depth	1	1	2	Avg.	1	2	≥ 3	Avg.	2	≥ 3	Avg.	Avg.
Closed-Source Language Models												
GPT-4-1106	0.881	**0.787**	0.759	0.766	**0.815**	**0.772**	**0.694**	**0.765**	0.802	0.626	0.675	**0.800**
Claude-3-Opus	**0.886**	0.784	**0.779**	**0.780**	0.764	0.749	0.592	0.724	0.695	0.576	0.609	0.788
GLM-4	0.868	0.763	0.739	0.745	0.768	0.739	0.626	0.724	**0.809**	**0.647**	**0.692**	0.779
ERNIEBot-4	0.866	0.749	0.735	0.738	0.725	0.696	0.649	0.692	0.756	0.600	0.643	0.764
GPT-3.5-Turbo-1106	0.845	0.686	0.630	0.644	0.661	0.561	0.475	0.561	0.565	0.482	0.505	0.682
Open-Source Language Models												
Qwen1.5-72B-Chat	0.873	0.749	0.730	0.735	0.751	0.698	0.521	0.675	0.611	0.521	0.546	0.752
Llama-3-70B-Instruct	0.858	0.769	0.722	0.733	0.747	0.704	0.675	0.706	0.573	0.571	0.571	0.757
InternLM2-20B-Chat	0.796	0.666	0.648	0.652	0.648	0.599	0.543	0.597	0.611	0.488	0.522	0.678
Qwen1.5-14B-Chat	0.817	0.657	0.636	0.641	0.622	0.621	0.536	0.606	0.550	0.435	0.467	0.680
Baichuan2-13B-Chat	0.760	0.583	0.517	0.533	0.571	0.479	0.404	0.480	0.443	0.409	0.418	0.591
Llama-3-8B-Instruct	0.778	0.669	0.568	0.592	0.597	0.552	0.483	0.546	0.626	0.429	0.484	0.638
Mistral-7B-Instruct	0.737	0.574	0.556	0.560	0.554	0.493	0.411	0.488	0.534	0.374	0.418	0.592
Qwen1.5-7B-Chat	0.802	0.598	0.611	0.608	0.519	0.564	0.570	0.558	0.634	0.491	0.531	0.658
InternLM2-7B-Chat	0.755	0.633	0.598	0.607	0.532	0.568	0.525	0.555	0.550	0.432	0.465	0.634
ChatGLM3-6B-Chat	0.701	0.556	0.490	0.506	0.455	0.430	0.411	0.431	0.573	0.312	0.384	0.546

Table 5: DRFR of LLMs computed by our proposed RAL method. The highest performance among open-source models is underlined, while the highest performance overall is **bold**.

Experiment: Main Results

- The performance of all LLMs declines with an increase in the complexity of composition types, especially on *Selection* and *Chain*
- The performance of most open-source LLMs falls short compared to closed-source LLMs, especially on complex composition types
- LLMs perform variously under different constraints and composition types.
  - For constraints, those having explicit evaluation standards, such as Format and Lexical, prove to be more challenging for LLMs
  - For compositions, *Chain* presents severe challenges while *Selection* comes second

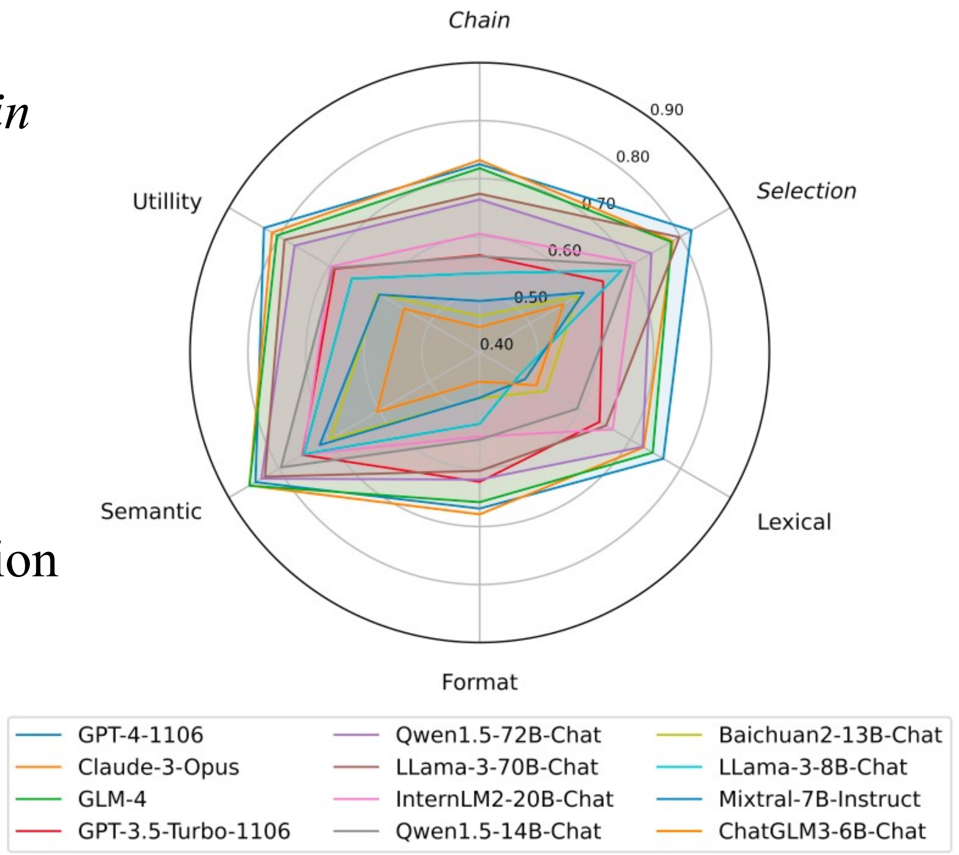


Figure 6: The performance of LLMs on different constraint and composition types.

Experiment: Analysis

- Decomposing complex instructions and executing them through multi-round interactions can not improve the performance of LLMs

Category	Nesting Depth	Origin	Decomposition	Δ
And	1	0.845	0.845	0.000
Chain	1	0.686	0.655	-0.031
	2	0.630	0.583	**-0.047**
Selection	1	0.661	0.631	-0.030
	2	0.561	0.520	-0.041
	≥ 3	0.475	0.411	**-0.064**
Selection & Chain	2	0.565	0.504	-0.061
	≥ 3	0.482	0.415	**-0.067**
Overall	-	0.682	0.652	-0.030

Table 6: The performance of GPT-3.5-Turbo-1106 on original and decomposed instructions.

Experiment: Analysis

- ComplexBench can provide a complementary perspective for LLM evaluation.

Model	ComplexBench	IFEval	HumanEval	MATH
GPT-4-1106	0.800	75.4	84.6	64.3
GLM-4	0.779	66.7	72.0	47.9
Qwen1.5-72B-Chat	0.752	55.8	71.3	42.5
Llama-3-70B-Instruct	0.757	78.9	81.7	50.4
Llama-3-8B-Instruct	0.638	68.6	62.2	30.0
Mistral-7B-Instruct	0.592	40.5	30.5	13.1
Qwen1.5-7B-Chat	0.658	38.8	46.3	23.2
InternLM2-7B-Chat	0.634	46.5	59.8	23.0
ChatGLM3-6B-Chat	0.546	28.1	64.0	25.7
Correlation with ComplexBench	-	0.814	0.715	0.895

Table 7: Model comparison on different abilities. The last row shows the Pearson correlation between the performance of LLMs in ComplexBench and other benchmarks.

Thanks for your attention!

Paper: https://arxiv.org/abs/2407.03978

Code: https://github.com/thu-coai/ComplexBench

清華大學 Tsinghua University