StructEval: Deepen and Broaden Large Language Model Assessment via Structured Evaluation

Boxi Cao[1,3], **Mengjie Ren**[1,3], **Hongyu Lin**[1], **Xianpei Han**[1,2,4✉]
Feng Zhang[5], **Junfeng Zhan**[5], **Le Sun**[1,2,4✉]

[1]Chinese Information Processing Laboratory [2]State Key Laboratory of Computer Science
Institute of Software, Chinese Academy of Sciences, Beijing, China
[3]University of Chinese Academy of Sciences, Beijing, China
[4]Key Laboratory of System Software, Chinese Academy of Sciences
[5]ByteDance Inc.
{boxi2020,hongyu,xianpei,sunle}@iscas.ac.cn

Abstract

Evaluation is the baton for the development of large language models (LLMs). Current evaluations typically employ a single-item assessment paradigm for each atomic test objective, which struggles to discern whether a model genuinely possesses the required capabilities or merely memorizes/guesses the answers to specific questions. To this end, this paper proposes a novel evaluation framework referred to as StructEval. Starting from an atomic test objective, StructEval deepens and broadens the evaluation by conducting a structured assessment across multiple cognitive levels and critical concepts, and therefore offers a comprehensive, robust and consistent evaluation for LLMs. Experiments on three widely-used benchmarks demonstrate that StructEval serves as a reliable tool for resisting the risk of data contamination and reducing the interference of potential biases, thereby providing more reliable and consistent conclusions regarding model capabilities. Our framework also sheds light on the design of future principled and trustworthy LLM evaluation protocols[1].

1 Introduction

Evaluation is fundamental for the development of large language models (LLMs) (Ouyang et al., 2022; Touvron et al., 2023b; OpenAI, 2023), providing essential measurements, feedback, and insights that facilitate enhancements in helpfulness, reliability and security (Chang et al., 2023). Consequently, a variety of large-scale benchmarks are proposed to assess LLMs' capabilities, such as language understanding (Hendrycks et al., 2021; Huang et al., 2023a), instruction following (Li et al., 2023; Zheng et al., 2023a), reasoning capabilities (Cobbe et al., 2021; Srivastava et al., 2022a).

(a) Previous evaluation uses single-item assessment for each atomic test objective.

(b) StructEval evaluates each test objective across multiple cognitive levels and critical concepts.
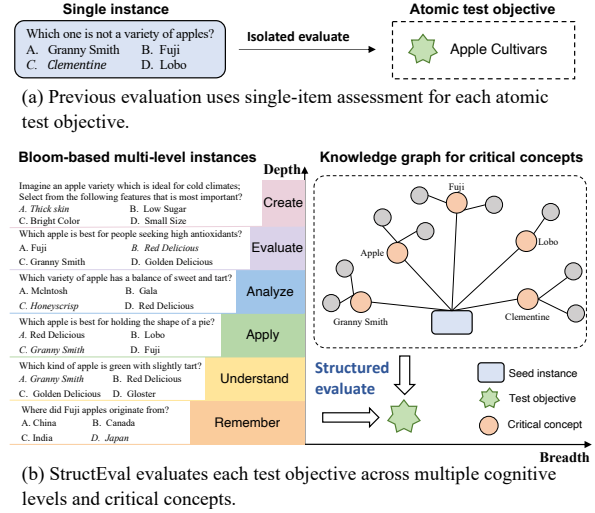
Figure 1: The illustrations for previous single-item assessment and our structured evaluation paradigm.

Unfortunately, current evaluations for LLMs typically employ a single-item assessment paradigm(Milton et al., 2011), which still suffer from their weakness on validity, robustness and comprehensiveness. As demonstrated in Figure 1a, to evaluate the factual knowledge in LLMs, they segment the factual knowledge into a set of atomic test objectives (e.g., *apple cultivars, function of insulin*), and evaluate each with a single instance (e.g., *which one is not a variety of apple*). However, such a single-item assessment paradigm struggles to discern whether a model genuinely possesses the required capability or merely memorizes/guesses the answers to specific questions. On the one hand, the single-item assessment relies on the correctness of isolated instances, which is sensitive to confounders correlated to specific instances (Poerner et al., 2020; Zhu et al., 2023b), and susceptible to biases or shortcuts (Cao et al., 2022; Xie et al., 2023; Wang et al., 2023a), making it difficult to discern whether a model's correct response is due to genuine understanding

---

[1]We openly release our source code and latest benchmark at https://github.com/c-box/StructEval, as well as leaderboard at https://huggingface.co/spaces/Bowieee/StructEval_leaderboard.

or mere memorization (Cao et al., 2021, 2024). On the other hand, the rapid expansion of LLMs' training data and memorization capacity have heightened the risk of data contamination in static benchmarks(Carlini et al., 2022; Jiang et al., 2024), potentially leading to inflated evaluations of model capabilities (Magar and Schwartz, 2022; Oren et al., 2023; Shi et al., 2023). That is, the true capabilities of the models might be overestimated owing to the potential contamination of the training dataset by test instances. Moreover, due to the huge resources required for benchmark construction, currently most benchmarks assess models in a static manner. Consequently, they may quickly reach saturation due to the inability to update in timeliness, complexity and diversity.

To address the aforementioned challenges, previous research has primarily attempted to manually construct newer, harder, and more diverse benchmarks. For instance, Kasai et al. (2022); Yu et al. (2023) devised evaluation benchmarks drawing from recent news or articles; Wang et al. (2021, 2023b) added perturbations into the original datasets to assess model robustness; Hendrycks et al. (2021); Huang et al. (2023a) collected test instances from human professional examination to increase difficulty and diversity. Despite the substantial resource invested, the single-item assessment paradigm of previous benchmarks still struggles with determining whether the evaluated performance can faithfully and fairly reflect the capabilities of models.

In this paper, we propose a novel structured evaluation framework named StructEval, which can comprehensively, robustly and validly evaluate LLMs. This is achieved by employing a structured assessment guided by pedagogy theories to evaluate model ability for each test objective across multiple cognitive levels and critical concepts, rather than relying on the correctness of a single test instance. Specifically, as illustrated in Figure 1b, StructEval consists of two modules which deepen and broaden current evaluation respectively. Given a seed instance, the first module identifies its underlying test objective, and then generates multiple test instances around this test objective which are aligned with the six cognitive levels outlined in Bloom's Taxonomy (Krathwohl, 2002). Meanwhile, the second module extracts the key concepts that must be understood to answer the seed question (Trochim, 1989), and then develop a series of instances revolving around these concepts based

on knowledge graph. Unlike single-item assessment, for each test objective, StructEval requires LLMs to demonstrate knowledge across multiple cognitive levels and a thorough comprehension of critical concepts for good performance. In this way, for each test objective, the assessment conclusion is no longer determined by the correctness of a single instance. As a result, it does not depend on confounders introduced by specific instances, such as prompt selection, surface form shortcut, data distribution, etc. Therefore, StructEval can reduce the impact of biases brought by these confounders, providing more consistent and accurate assessment conclusions for various LLMs. Meanwhile, a model with data contamination can merely memorize specific answers but still lacks corresponding structured knowledge, therefore, StructEval can robustly provide stable assessment results even when the training data is contaminated. Moreover, due to StructEval's capability to automatically generate large-scale and high-quality instances, thereby realizing dynamic evaluation through updating of knowledge sources, it can also prevent benchmarks from rapidly reaching saturation.

To demonstrate the effectiveness of our framework, we implement StructEval based on 3 widely used benchmarks. The experiments on a variety of LLMs demonstrate that StructEval: 1) enables the automating generation of large-scale benchmarks and completion of structured evaluations, while ensures instance correctness, relevance, and helpfulness. 2) effectively resists the risk of data contamination, providing robust evaluation results even under data contamination settings. 3) significantly enhances the consistency of model rankings across different experiments, offering more precise and stable conclusions from evaluations. 4) substantially outperforms previous augmentation-based strategies such as word perturbation, paraphrasing, back translation, option shuffle, etc.

The main contributions of this paper include:

- We propose a novel evaluation framework named StructEval, which can comprehensively evaluate LLMs' capability by assessing each test objective across multiple cognitive levels and critical concepts in principle, rather than previous single-item assessment.

- We implement StructEval on widely-used benchmarks, and human evaluation results demonstrate that StructEval can automatically construct large-scale benchmark with
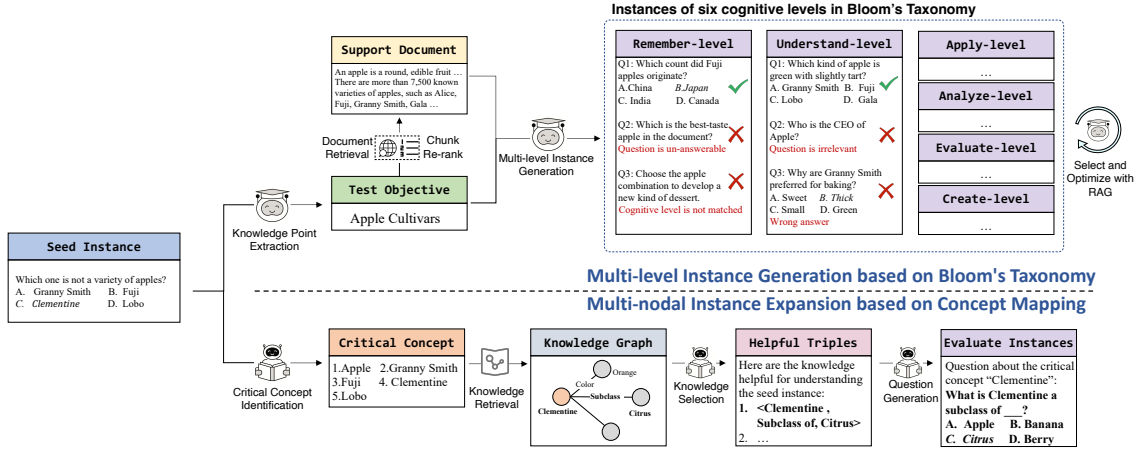
Figure 2: The illustration of StructEval framework, which consists of two modules. The first module aims to evaluate the model's ability on test objective across multiple cognitive levels in Bloom's Taxonomy. The second module aims to evaluate the model's understanding of relevant critical concepts based on knowledge graph.

high quality.

- We conduct comprehensive experiments regarding data contamination and rank consistency, demonstrating the effectiveness, robustness and consistency of StructEval for LLM evaluation.

2 Preliminaries

Evaluation is the cornerstone for the progress of LLMs (Chang et al., 2023). Unfortunately, there still exist several grand challenges for achieving comprehensive and trustworthy evaluation for LLMs. For instance, the inability to scale in complexity and diversity at the same pace as the rapid advancements in model capabilities(Srivastava et al., 2022b; Huang et al., 2023b); the biases or shortcuts that lead to unfaithful assessments (Liang et al., 2022; Xie et al., 2023); and the lack of reliable metrics for providing trustworthy results (Zheng et al., 2023b; Wang et al., 2023c). To this end, previous studies have mainly devoted to improving the diversity, scale, difficulty and timeliness of test instances (Kasai et al., 2022; Zhu et al., 2023a), exploring the robustness and trustworthiness vulnerabilities in current evaluations (Zhu et al., 2023b; Wang et al., 2023a), and proposing metrics or protocols more suitable for generative LMs (Lin and Chen, 2023; Zhang et al., 2023). In comparison, this paper aims to propose a structured evaluation framework for LLM evaluation.

StructEval framework is guided by two pedagogy theories which are widely used for educational assessment. *Bloom's Taxonomy Theory* is a

hierarchical model[2] used for classification of educational learning objectives into six levels, including remember, understand, apply, analysis, evaluate and create (Krathwohl, 2002). Therefore, to comprehensively evaluate the model's ability across various cognitive levels on test objective, StructEval would generate multiple test instances covering six cognitive levels in Bloom's Taxonomy. *Concept Mapping theory* is another well-known tool for student assessment. Educators use concept maps to assess the breadth of a student's understanding of a subject, which reveals how well students grasp connections among concepts (Trochim, 1989). Therefore, to assess whether the model genuinely possess the knowledge required for test instance, StructEval would develop a series of instances revolving the critical concepts based on knowledge graph.

3 StructEval Framework

The overall framework of StructEval is illustrated in Figure 2, which consists of two modules. Given a seed instance, the first module would evaluate the model's ability on test objective across multi cognitive level. It first identifies the underlying test objective of this instance and then generates multiple relevant instances covering six cognitive levels of Bloom's Taxonomy. The second module evaluates the model's comprehensive understanding of all critical concepts related to the seed instance. It extracts the essential concepts that must be understood and develops a series of extended questions around these concepts using a knowledge graph.

---

[2]We adopt the cognitive domain list since it is frequently used for educational assessment.

In the following, we would describe the StructEval framework in detail.

3.1 Bloom's Taxonomy-based Instance Generation

As shown in Figure 2, given a seed instance, the first module of StructEval automatically generates test instances corresponding to the six cognitive levels in Bloom's Taxonomy through the following steps: 1) extract the test objective examined by the seed instance; 2) retrieve relevant documents and re-rank the document chunks based on their relevance to seed instance; 3) generate candidate evaluation instances for each cognitive level in Bloom's Taxonomy using in-context learning; 4) select instances that best meet the requirements and refine them to be more challenging. Subsequently, each component will be introduced in detail.

Test Objective Extraction aims to identify the underlying test objective for each seed instance. For example, the test objective for question "*Which one is not a variety of apple?*" is "*apple cultivars*". However, such a single question is insufficient to thoroughly evaluate the LLM's related knowledge. Therefore, to comprehensively assess how much knowledge the LLM possesses about the test objective and its level of understanding across different cognitive tiers, we conduct a structured evaluation around this test objective. In our framework, we prompt LLM with few-shot demonstration to extract the test objective examined by each instance in the benchmarks.

Relevant Document Retrieval Given the test objective corresponding to the seed instance, an intuitive approach is directly prompting LLM to generate instances for each cognitive level. However, this approach is severely compromised by the LLM's hallucinations, resulting in a significant proportion of incorrect instances. Therefore, StructEval would first retrieve relevant passages, and then re-rank document chunks based on the correlation with the seed instance. This procedure ensures that the generation of subsequent instances is firmly based on the retrieved context, guaranteeing the precision and pertinence of the generated instances.

Candidate Instances Generation aims to generate multiple candidate instances for each cognitive level in Bloom's Taxonomy, based on the test objective with relevant document chunks. As

> # Background
> Bloom's Taxonomy categorizes educational objectives into six levels. Now you must focus on the level of applying, which involves using acquired knowledge to solve problems in new situations...
>
> # Instruction
> For a given test objective and document, generate 5 instances based on the following principles:
> 1. Ensure the question can be answered independently without additional context.
> 2. Ensure the correct answer and supporting evidence are available within the provided document.
> 3. Ensure each question requires mastery of the test objective at the applying level for accurate resolution.
>
> **Few shot demonstrations**
>
> # User Input
> test objective: <objective>
> Relevant Documents: <documents>
> 5 instances of applying level in Bloom's Taxonomy:

Table 1: Prompt design for candidate instances generation. Please refer to the Appendix C for more framework design details.

demonstrated in Table 1, we meticulously design the prompt for LLM to generate relevant, correct and helpful instances corresponding to each cognitive level. The prompt begins with introducing Bloom's Taxonomy and current cognitive level, following by the task instruction which includes three principles to ensure the answerability, accuracy and relevance of the generated instances. Subsequently, we provide manually created few-shot demonstrations, and ask LLM to generate candidate instances using these demonstrations as references.

Instance Selection and Refinement Since the quality and difficulty of these instances may vary greatly, as shown in Figure 2, we introduce a post-processing module aimed at selecting the highest quality instances for each cognitive level. 1) To ensure the answerability and correctness of instances, we prompt LLM to eliminate questions necessitating specific contextual information for resolution, and employ Retrieval-Augmented Generation (RAG) module to exclude questions that cannot be correctly answered based on the provided context, thereby ensuring the accuracy of the generated answers; 2) To enhance the quality and difficulty of instances, inspired by Clark et al. (2018); Lin et al. (2022), we establish a comprehensive pool of diverse LMs. Questions that all models could answer correctly were eliminated, thus ensuring

the discriminative efficacy.

Ultimately, for each instance within the original benchmark, we develop a hierarchical evaluation system capable of extensively assessing the tested model's knowledge across all six cognitive levels in Bloom's Taxonomy.

3.2 Concept Mapping-based Instance Expansion

The second module evaluates LLMs' knowledge for each test objective with a concept map. The hypothesis behind is also intuitive: if a model genuinely possesses the necessary knowledge to answer a given question, it should demonstrate a comprehensive understanding of the critical relevant concepts. Specifically, as illustrated in Figure 2, StructEval utilizes LLM and knowledge graph to expand the breadth of existing benchmarks with following steps: 1) Identify the key concepts that must be understood to correctly answer the seed question; 2) Retrieve relevant knowledge sub-graphs for each concept and select the necessary knowledge triplets from all the candidates for understanding the original question; 3) Transform the selected triplets into test instances and optimize their difficulty.

Critical Concept Identification   aims to identify the critical concepts that must be understood to correctly answer the seed question. These concepts are then linked to the entries in knowledge graph to facilitate subsequent knowledge retrieval. Previous approaches such as BLINK (Wu et al., 2020) are constrained on the entity label set and fail to discern between critical and non-critical concepts. Therefore, we prompt LLM with few-shot demonstration to identify the critical concepts in instances.

Knowledge Graph Retrieval and Selection   involves retrieving the identified critical concepts across the entire knowledge graph and extracting relevant knowledge triples from the sub-graph as candidates. Given the potential enormity of the candidate set, which may contain extraneous triplets not aiding in determining the model's ability to answer the seed question, similar to Guan et al. (2023), we prompt the LLM to select the helpful knowledge triplets with few-shot demonstrations.

Instances Generation and Optimization   transforms the selected factual triples into evaluation instances. Similar to Petroni et al. (2019), we utilize the subject entity and its relation to formulate

the question, with the object entity as the answer. For multiple-choice questions, in order to ensure the difficulty of the questions, we first use the taxonomy of the knowledge graph to identify the finest-grained entity category corresponding to the correct answer. Then, we select the incorrect options from other entities within the same category.

Ultimately, we construct a multi-nodal evaluation framework for each test instance, offering a comprehensive assessment of the language model's grasp of pertinent critical concepts.

4 Implementations and Experiments of StructEval

In this section, we first implement StructEval across three widely-used benchmarks. Through human evaluation, we demonstrate capability of StructEval to automatically construct large-scale benchmarks while ensuring the helpfulness, answerability and correctness of generated instances. Then, we demonstrate how StructEval could improve the robustness and consistency of LLM evaluation from the following perspectives. Firstly, StructEval requires LLMs to understand the test objective across multiple cognitive levels and critical concepts. In this case, a contaminated model which merely memorize specific answers may achieve high performance in original benchmark, but cannot gain performance improvements on the structured evaluation since it lacks of corresponding knowledge. Therefore, StructEval can effectively resist data contamination issues, providing robust evaluation results even when the test data is leaked. Secondly, since the evaluation results do not rely on the correctness on single instance, it does not depend on confounders introduced by specific instances, such as prompt selection, surface form shortcut and data distribution. Therefore, compared with single-item assessment, StructEval can provide a much more robust and consistent evaluation conclusion.

4.1 StructEval -based Benchmarks

Finding 1. *By leveraging the advanced generative capabilities of LLMs, and meticulously orchestrating the construction process guided by pedagogy theories and grounded in credible knowledge sources, StructEval is capable of automatically construct large-scale benchmarks while ensuring the helpfulness, answerability and accuracy of generated instances.*

To demonstrate the reliability and quality of the automatically generated instances by StructEval, we adopt StructEval on three widely used benchmarks for LLMs, and conduct human evaluations from three aspects.

Seed Benchmarks include the following three wildly used LLM evaluation benchmarks, and the corresponding statistics are demonstrated in Table 2: 1) *MMLU* (Hendrycks et al., 2021) is a large-scale benchmark designed to measure the knowledge in large language models. We filtered out subjects unsuitable for StructEval, such as logical reasoning and numerical computation, and used the remaining 48 subjects for experiments. 2) *ARC* (Clark et al., 2018) consists of a set of science exam questions drawn from a variety of sources, which is widely used to assess the knowledge in LLMs. The benchmark is partitioned into a challenge set and easy set, and we include both of them in our experiments. 3) *OpenBook QA* (Mihaylov et al., 2018) is a question-answering datasets that consists of multiple-choice elementary-level science questions. For each benchmark, we randomly sample 200 test instances generated by StructEval to conduct human evaluations.

LLM and Knowledge Source Implementation Considering the balance between cost, efficiency, and the quality of generation, we select ChatGPT[3] for LLM generation tasks in this paper, and our framework can also be easily adopted to other large models and knowledge sources. We adopt BM25 (Maron and Kuhns, 1960) for document retrieval and BGE (Xiao et al., 2023) for chunk re-ranking. We select Wikipedia (Wikipedia contributors, 2004) as knowledge source since it encompasses the vast majority information about test objectives and possesses an high density of knowledge, and use Wikidata (Vrandečić and Krötzsch, 2014) for fact retrieval, since it is one of the most comprehensive knowledge bases covering structured knowledge.

Metrics We propose the following metrics to comprehensively assess the quality of test instances generated through StructEval: 1) *Instance Helpfulness*, which is calculated by the proportion of generated evaluation instances that conform to the target test objective or critical related concepts;

---

[3]Gpt-3.5-turbo-1106 is used in the experiments of this paper, and GPT-4o-mini is used for our latest benchmark construction.

Benchmark	Original	Bloom	Concept	Structured
MMLU	13.1k	135.8k	33.1k	168.9k
ARC-Easy	2.3k	17.8k	6.2k	24.0k
ARC-Challenge	1.1k	14.2k	4.2k	18.4k
OpenBook QA	0.5k	6.0k	1.2k	7.2k

Table 2: Data statistics of the seed benchmarks and StructEval -constructed benchmarks. "Bloom" denotes the multi-level instances based on Bloom's taxonomy, "Concept" denotes the test instances about critical concepts, "Structured" indicates the comprehensive structured assessment that includes both components.

Benchmark	Helpfulness	Answerability	Correctness
Struct MMLU	95.5	96.0	97.0
Struct ARC	96.0	96.0	98.0
Struct OpenBook	97.5	95.0	94.5

Table 3: The human evaluation results three benchmarks constructed by StructEval.

2) *Question Answerability*, which is calculated by the proportion of generated questions that can be answered without relying on external context. 3) *Answer Correctness*, which is calculated by the proportion of generated evaluation instances that contain the correct answer. The detailed annotation guideline is presented in the appendix.

Benchmark Assessment From table 2, we find that StructEval is able to automatically construct a large-scale multi-level and multi-nodal evaluation system based on original benchmark, provide novel test instances and structured evaluation protocol for existing benchmarks. Moreover, the human evaluation results are shown in table 3. We can find that StructEval could construct the structured evaluations while ensure the high quality of generated instances in all the aspects of instance helpfulness, question answerability, and answer correctness. The few errors mainly due to that GPT-3.5 generate questions relying on context to answer or with multiple correct choices, we also provide a detailed error analysis and the annotation guideline in the Appendix E due to page limitations.

4.2 Robustness of StructEval

Finding 2. *By expanding the benchmark across both depth and breadth dimensions, StructEval is capable of robustly evaluating the capabilities of LLMs, resisting the risks of data contamination, and providing stable results even under data contamination settings.*

Data contamination refers to the inclusion of test data in the training dataset of evaluated models,

Method	LLaMa-7B			LLaMa-30B			LLaMa-2-7B			LLaMa-2-13B			Mistral-7B		
	Clean IFT	w/ Test ⇓	Δ ⇓	Clean IFT	w/ Test ⇓	Δ ⇓	Clean IFT	w/ Test ⇓	Δ ⇓	Clean IFT	w/ Test ⇓	Δ ⇓	Clean IFT	w/ Test ⇓	Δ ⇓
MMLU															
Original	50.22	79.32	+29.10	59.06	92.22	+33.16	54.42	86.88	+32.46	59.27	90.98	+31.71	55.84	95.67	+39.83
CharDisturb	49.52	76.90	+27.38	58.38	90.76	+32.38	53.41	83.99	+30.58	57.88	88.86	+30.98	55.28	94.04	+38.76
Wordnet	49.07	75.76	+26.69	57.80	90.11	+32.31	53.58	83.25	+29.67	58.06	88.16	+30.10	54.95	93.38	+38.43
Paraphrasing	50.19	73.09	+22.84	58.74	89.33	+30.59	54.50	80.76	+26.26	58.20	86.05	+27.85	55.48	92.79	+37.31
BackTranslation	50.36	76.86	+26.50	58.71	90.95	+32.24	54.22	84.21	+29.99	59.23	89.27	+30.04	55.95	94.53	+38.58
OptionShuffle	50.70	66.50	+15.80	58.69	83.69	+25.00	53.84	72.57	+18.73	58.94	78.04	+19.10	54.97	84.21	+29.24
StructEval (ours)	47.60	49.33	**+1.73**	56.87	57.09	**+0.22**	51.42	53.24	**+1.82**	56.53	57.32	**+0.79**	52.28	53.95	**+1.67**
ARC-challenge															
Original	53.86	91.77	+37.91	69.55	98.54	+28.99	61.75	96.83	+35.08	72.04	98.97	+26.93	66.72	99.83	+33.11
CharDisturb	52.47	88.99	+36.52	68.26	97.27	+29.01	58.19	94.71	+36.52	69.88	97.27	+27.39	64.68	99.15	+34.47
Wordnet	50.51	86.18	+35.67	65.36	95.05	+29.69	58.36	92.75	+34.39	67.75	96.33	+28.58	61.95	97.27	+35.32
Paraphrasing	52.56	82.00	+29.44	68.69	95.39	+26.70	59.04	89.51	+30.47	69.34	93.34	+23.20	64.51	96.76	+32.25
BackTranslation	53.84	88.05	+34.21	66.98	97.27	+30.29	60.84	94.97	+34.13	69.80	97.44	+27.64	64.85	98.55	+33.70
OptionShuffle	53.84	76.37	+22.53	70.22	93.26	+23.04	62.12	83.36	+21.24	70.05	90.02	+19.97	65.44	94.45	+29.01
StructEval (ours)	44.90	44.71	**-0.19**	54.49	55.23	**+0.74**	48.96	49.69	**+0.73**	54.58	55.40	**+0.82**	52.56	52.27	**-0.29**
ARC-easy															
Original	77.06	97.22	+20.16	86.95	99.49	+12.54	79.38	99.16	+19.78	85.40	99.62	+14.22	81.73	99.96	+18.23
CharDisturb	74.24	94.70	+20.46	85.40	98.19	+12.79	76.52	97.31	+20.79	79.17	99.28	+20.11	79.17	99.28	+20.11
Wordnet	73.40	93.60	+20.20	83.84	97.10	+13.26	75.08	95.58	+20.50	81.52	96.93	+15.41	78.37	98.06	+19.69
Paraphrasing	75.21	93.01	+17.80	86.24	97.43	+11.19	78.24	95.33	+17.09	83.33	96.80	+13.47	81.69	98.65	+16.96
BackTranslation	75.84	95.12	+19.28	84.72	98.06	+13.34	77.27	97.31	+20.04	83.16	98.19	+15.03	80.26	99.12	+18.86
OptionShuffle	75.72	90.74	+15.02	86.20	96.76	+10.56	78.41	93.39	+14.98	85.27	96.21	+10.94	81.52	98.15	+16.63
StructEval (ours)	45.21	45.37	**+0.16**	54.14	55.34	**+1.20**	48.60	49.60	**+1.00**	54.25	54.95	**+0.70**	51.23	51.79	**+0.56**
OpenBook QA															
Original	55.86	87.03	+31.17	56.11	97.51	+41.40	62.09	95.76	+33.67	66.83	99.50	+32.67	69.58	99.00	+29.42
CharDisturb	48.40	82.20	+33.80	51.20	95.00	+43.80	53.40	89.20	+35.80	60.80	94.60	+33.80	61.80	97.20	+35.40
Wordnet	49.00	79.40	+30.40	52.20	92.60	+40.40	55.20	86.80	+31.60	57.40	93.40	+36.00	61.00	96.60	+35.60
Paraphrasing	59.00	77.20	+18.20	57.20	92.80	+35.60	59.80	84.80	+25.00	67.60	93.20	+25.60	68.20	95.60	+27.40
BackTranslation	51.80	84.80	+33.00	52.00	95.00	+43.00	58.80	90.40	+31.60	62.40	96.00	+33.60	65.60	97.00	+31.40
OptionShuffle	53.60	87.60	+34.00	53.80	97.60	+43.80	59.40	95.40	+36.00	64.60	99.60	+35.00	67.60	98.80	+31.20
StructEval (ours)	44.31	42.57	**-1.74**	51.97	51.71	**-0.26**	48.86	46.89	**-1.97**	55.48	55.87	**+0.39**	51.00	49.64	**-1.36**

Table 4: Performance comparisons of LLMs which are trained on clean data and contaminated data. "w/Test" indicates that the instruction tuning data is contaminated by the test samples. "Δ" denotes the performance divergence between clean and contamination settings, lower values (⇓) reflect that the corresponding evaluation is less affected by data contamination. The evaluation is conducted under zero-shot setting.

which can significantly skew the apparent performance and capabilities of models, leading to misleading conclusions about their true effectiveness. Addressing data contamination becomes increasingly crucial for large language models as the training data grow exponentially with the data sources and processing recipes being obscure.

To demonstrate the effectiveness of StructEval in resisting the risk of data contamination, we compare the performance divergences of LLMs with and without data contamination, on the original benchmark, the data augmented benchmark and StructEval-constructed benchmark respectively. Specifically, for a seed benchmark and a base model, we use instruction fine-tuning (IFT) to train the model on both a clean dataset and a dataset contaminated with test data. To make a fair comparison, we ensure that both datasets maintain identical scale and similar data composition. Simultaneously, we integrate Alpaca-GPT-4 (Taori et al., 2023) dataset into both the training data to ensure data diversity and prevent training collapse. In this case, the **contaminated set** consists of Alpaca-GPT-4 and the test data, while the **clean set** consists of Alpaca-GPT-4 and an equal number of multi-choice questions which are randomly sampled from an out-of-distribution benchmark Xiezhi (Gu et al.,

2023). To ensure the robustness of our conclusions, we consider 5 wildly used base LLMs of various scales including LLaMa-7B&30B(Touvron et al., 2023a), LLaMa-2-7B&13B(Touvron et al., 2023b) and Mistral-7B(Jiang et al., 2023). Each model is trained through 3 epochs with batch size of 256 sequences, using Adam with learning rate $2e-5$.

We also compare our method with the following augmentation-based approaches including character-level, word-level and instance-level, which are able to generate adversarial samples while ensuring the answerability and correctness of test instances: 1) CharDisturb (Morris et al., 2020): which randomly substitutes, deletes, inserts and swaps characters in original question. 2) Word-Net (Miller, 1992), which randomly replaces words with WordNet synonyms. 3) Paraphrasing (Zhu et al., 2023c), which prompts ChatGPT to generate paraphrasing for each test question. 4) BackTranslation (Sennrich et al., 2016), which translates the test question into another language and translates it back. 5) OptionShuffle (Yang et al., 2023), which re-ordered the options for each question to prevent LLMs memorizing specific option for question.

The results in Table 4 clealy demonstrate the significant role of StructEval in resisting data contamination: 1) **The performance of original bench-**

Model	Ori	Char	Word	Trans	Shuffle	Struct
Mistral-7B⋆	59.74	57.09	54.32	45.35	42.08	**61.86**
Mistral-8*7B†	46.73	50.38	52.16	55.40	66.05	**100.00**
LLaMA-7B⋆	99.91	99.95	99.14	99.89	95.58	**99.99**
LLaMA-30B⋆	38.97	44.84	43.42	41.58	41.90	**70.32**
LLaMA-2-7B⋆	37.31	40.02	41.77	**59.31**	36.07	48.39
LLaMA-2-13B⋆	38.29	49.63	44.20	46.64	48.89	**76.80**
Baichuan2-7B†	45.37	49.93	85.71	67.10	57.89	**94.42**
Baichuan2-13B†	55.32	41.78	58.00	**64.70**	40.98	53.93
Qwen-7B†	40.09	40.70	48.65	47.32	32.59	**94.41**
Qwen-14B†	87.18	**88.27**	84.33	84.88	71.44	86.86
Qwen1.5-7B†	51.55	40.46	42.84	49.03	34.27	**98.59**
Qwen1.5-14B†	57.84	59.95	52.54	55.35	64.26	**100.00**
Yi-6B†	59.76	68.68	60.49	40.91	52.05	**100.00**
Overall Rank	1.24	1.63	3.15	2.28	1.48	**33.17**

Table 5: The *rank consistency* of each LLM over 10000 task samples, and each task consists of $K = 15$ subjects from MMLU. "Overall Rank" indicates percentage of the most popular rank of all models across 10000 runtimes. "⋆" denote that the base model is trained with clean IFT. "†" denote the chat version of model.

mark can severely suffer from data contamination due to the superior memorizing capabilities of LLMs, resulting in a serious overestimation of the model's capabilities. For instance, the performance of all models on MMLU increase by at least 29% when the training data is contaminated 2) **Previous augmentation-based approaches struggle to resist data contamination.** Despite adjustments to the surface form of the original instances, due to the LLMs' advanced memorizing and language comprehension capabilities, they still achieve significant benefits from data contamination. 3) **StructEval is able to provide stable evaluation results, regardless of whether the training data is contaminated.** For example, due to data contamination, the performance of LLaMa-2-13B improves by 31.71% on the original MMLU, but changes by only 0.79% on the structured-MMLU generated by StructEval, which remains almost unchanged. The finding remains consistent across all base LLMs and benchmarks. Such results effectively demonstrate that StructEval can play a role in anti-attack and contamination monitoring for evaluation.

4.3 Consistency of StructEval

Finding 3. *By conducting structured assessments across various cognitive levels and essential concepts, instead of basing assessments solely on the accuracy of a single instance, StructEval achieves valid assessment of models, providing consistent conclusions regarding various model capabilities.*

As we discussed above, StructEval can also serve as a more stable reference for assessing the knowledge capabilities of language models,
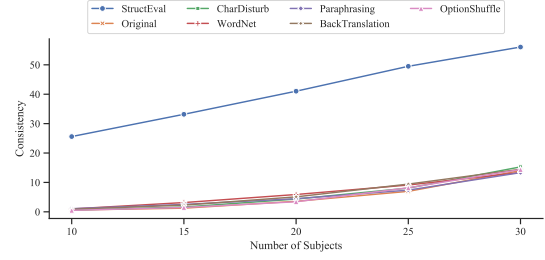

Figure 3: The comparison of *overall rank consistency* for each method. StructEval substantially outperforms original benchmark and all augmentation-based strategies as number of sampled subjects K changes.

which can give more stable evaluation results to various LLMs, and reach a consistent conclusion. Demonstrating this requires to collect numerous benchmarks with similar evaluation objective and distribution, and observe whether the evaluation conclusions are consistent on original data, augmented data and StructEval-constructed data respectively. To facilitate our experiments, we refer to Cao et al. (2022) and use *rank consistency* across multiple runtimes as the evaluation metric. Specifically, we randomly sample 10000 sub-set with K subjects from MMLU, and evaluate rank consistency by measuring the percentage of the most popular rank of each model in 10000 runtimes. For instance, if ChatGPT ranks at 3rd place in 6500 of the 10000 runtimes, then the rank consistency of ChatGPT would be 65%. To make a comprehensive evaluation, we conduct experiments on 13 different open-source large language models across various parameter scales, including LLaMA-7B&30B, LLaMA-2-7B&13B, Mistral-7B&8*7B, Baichuan2-7B&13B(Baichuan, 2023), Qwen-7B&14B, Qwen1.5-7B&14B(Bai et al., 2023) and Yi-6B. We report the rank consistency of each model, as well as the rank consistency across all models.

The results in Table 5 and Figure 3 demonstrate that StructEval can significantly improve the evaluation consistency: 1) **The consistency of current LLM evaluations are relatively poor**: when using original isolated instances to compare the ability of different models, the overall rank consistency is only 1.24%. 2) **Previous strategies can hardly improve the rank consistency.** Although they modify the original data, they still adhere to the paradigm of single-item assessment. As a result, they remain susceptible to interference from confounders and struggle to provide more consistent evaluation

conclusions across all models. 3) **StructEval provides much more consistent evaluation conclusions regarding the ability of different LLMs**: the overall rank consistency improved from 1.24% to 33.17% when $K = 15$, and the rank consistency of most LLM is substantially improved, reaching a more reliable conclusion.

5   Conclusion

This paper proposes a novel evaluation framework for large language models named StructEval. Through structurally evaluating model's capability for each test objective across multiple cognitive levels and critical concepts, StructEval achieves more comprehensive, robust and consistent evaluation for LLMs. Experimental results demonstrate that StructEval could effectively resist the risk of data contamination and significantly improve the rank consistency across models. The corresponding benchmarks and leaderboard will be released, which will benefit our understanding of LLMs' capabilities. StructEval is also broadly applicable to various applications. For instance, StructEval can function as a customizable benchmark construction framework, capable of automating evaluations for any granularity of assessment objectives, please refer to Appendix A for details and experiments. Furthermore, our study also sheds light on the design of future principled and trustworthy instance collection and LLM evaluation protocols.

Limitations

Considering the balance between cost, efficiency and quality for benchmark construction, we currently use GPT-3.5 for generation in this paper, which may limit the difficulty and quality of generated instances. In the future, we will introduce more powerful LLMs (e.g., GPT-4) or incorporate human to our framework, to further improve the qualify of test instances, and release the corresponding updated benchmarks. Moreover, to facilitate the assessment of our framework, we currently select to implement StructEval based on multi-choice benchmarks. Please also kindly note that our framework can be easily adapted to other formats of benchmark such as open-end QA and multi-turn conversation, which will be included in our future work.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Baichuan. 2023. Baichuan 2: Open large-scale language models. *arXiv preprint arXiv:2309.10305*.

Boxi Cao, Hongyu Lin, Xianpei Han, Fangchao Liu, and Le Sun. 2022. Can prompt probe pretrained language models? understanding the invisible risks from a causal view. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5796–5808, Dublin, Ireland. Association for Computational Linguistics.

Boxi Cao, Hongyu Lin, Xianpei Han, Le Sun, Lingyong Yan, Meng Liao, Tong Xue, and Jin Xu. 2021. Knowledgeable or educated guess? revisiting language models as knowledge bases. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1860–1874, Online. Association for Computational Linguistics.

Boxi Cao, Qiaoyu Tang, Hongyu Lin, Shanshan Jiang, Bin Dong, Xianpei Han, Jiawei Chen, Tianshu Wang, and Le Sun. 2024. Retentive or forgetful? diving into the knowledge memorizing mechanism of language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14016–14036, Torino, Italia. ELRA and ICCL.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2022. Quantifying Memorization Across Neural Language Models. *ArXiv preprint*, abs/2202.07646.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. 2023. A Survey on Evaluation of Large Language Models. *ArXiv preprint*, abs/2307.03109.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have Solved Question Answering? Try ARC, the AI2 Reasoning Challenge. *ArXiv preprint*, abs/1803.05457.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training Verifiers to Solve Math Word Problems. *ArXiv preprint*, abs/2110.14168.

Zhouhong Gu, Xiaoxuan Zhu, Haoning Ye, Lin Zhang, Jianchen Wang, Sihang Jiang, et al. 2023. Xiezhi: An Ever-Updating Benchmark for Holistic Domain Knowledge Evaluation. *ArXiv preprint*, abs/2306.05783.

Xinyan Guan, Yanjiang Liu, Hongyu Lin, Yaojie Lu, Ben He, Xianpei Han, and Le Sun. 2023. Mitigating Large Language Model Hallucinations via Autonomous Knowledge Graph-based Retrofitting. *ArXiv preprint*, abs/2311.13314.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, Yao Fu, Maosong Sun, and Junxian He. 2023a. C-Eval: A Multi-Level Multi-Discipline Chinese Evaluation Suite for Foundation Models. *ArXiv preprint*, abs/2305.08322.

Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, Yao Fu, Maosong Sun, and Junxian He. 2023b. C-Eval: A Multi-Level Multi-Discipline Chinese Evaluation Suite for Foundation Models. *ArXiv preprint*, abs/2305.08322.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B. *ArXiv preprint*, abs/2310.06825.

Minhao Jiang, Ken Ziyu Liu, Ming Zhong, Rylan Schaeffer, Siru Ouyang, Jiawei Han, and Sanmi Koyejo. 2024. Investigating Data Contamination for Pre-training Language Models. *ArXiv preprint*, abs/2401.06059.

Jungo Kasai, Keisuke Sakaguchi, Yoichi Takahashi, Ronan Le Bras, Akari Asai, Xinyan Yu, Dragomir Radev, Noah A. Smith, Yejin Choi, and Kentaro Inui. 2022. RealTime QA: What's the Answer Right Now? *ArXiv preprint*, abs/2207.13332.

David R. Krathwohl. 2002. A Revision of Bloom's Taxonomy: An Overview. *Theory Into Practice*, 41(4):212–218.

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

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, et al. 2022. Holistic Evaluation of Language Models. *ArXiv preprint*, abs/2211.09110.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Yen-Ting Lin and Yun-Nung Chen. 2023. LLM-Eval: Unified Multi-Dimensional Automatic Evaluation for Open-Domain Conversations with Large Language Models. *ArXiv preprint*, abs/2305.13711.

Inbal Magar and Roy Schwartz. 2022. Data contamination: From memorization to exploitation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 157–165, Dublin, Ireland. Association for Computational Linguistics.

M. E. Maron and J. L. Kuhns. 1960. On Relevance, Probabilistic Indexing and Information Retrieval. *Journal of the ACM*, 7(3):216–244.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.

George A. Miller. 1992. WordNet: A lexical database for English. In *Speech and Natural Language: Proceedings of a Workshop Held at Harriman, New York, February 23-26, 1992*.

K. Milton, F. C. Bull, and A. Bauman. 2011. Reliability and validity testing of a single-item physical activity measure. *British Journal of Sports Medicine*, 45(3):203–208.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. Textattack: A framework for adversarial attacks, data augmentation, and adversarial training in nlp. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126.

OpenAI. 2023. GPT-4 Technical Report. *ArXiv preprint*, abs/2303.08774.

Yonatan Oren, Nicole Meister, Niladri Chatterji, Faisal Ladhak, and Tatsunori B. Hashimoto. 2023. Proving Test Set Contamination in Black Box Language Models. *ArXiv preprint*, abs/2310.17623.

Long Ouyang, Jeff 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 Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. *ArXiv preprint*, abs/2203.02155.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Nina Poerner, Ulli Waltinger, and Hinrich Schütze. 2020. E-BERT: Efficient-yet-effective entity embeddings for BERT. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 803–818, Online. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. 2023. Detecting Pretraining Data from Large Language Models. *ArXiv preprint*, abs/2310.16789.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, et al. 2022a. Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models. *ArXiv preprint*, abs/2206.04615.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, et al. 2022b. Beyond the Imitation Game: Quantifying and extrapolating the capabilities of language models. *ArXiv preprint*, abs/2206.04615.

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, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. LLaMA: Open and Efficient Foundation Language Models. *ArXiv preprint*, abs/2302.13971.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, et al. 2023b. Llama 2: Open Foundation and Fine-Tuned Chat Models. *ArXiv preprint*, abs/2307.09288.

William M. K. Trochim. 1989. An introduction to concept mapping for planning and evaluation. *Evaluation and Program Planning*, 12(1):1–16.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: A free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Boxin Wang, Weixin Chen, Hengzhi Pei, Chulin Xie, Mintong Kang, Chenhui Zhang, Chejian Xu, Zidi Xiong, Ritik Dutta, Rylan Schaeffer, Sang T. Truong, Simran Arora, Mantas Mazeika, Dan Hendrycks, Zinan Lin, Yu Cheng, Sanmi Koyejo, Dawn Song, and Bo Li. 2023a. DecodingTrust: A Comprehensive Assessment of Trustworthiness in GPT Models. *ArXiv preprint*, abs/2306.11698.

Boxin Wang, Chejian Xu, Shuohang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Hassan Awadallah, and Bo Li. 2021. Adversarial GLUE: A Multi-Task Benchmark for Robustness Evaluation of Language Models. *ArXiv preprint*, abs/2111.02840.

Jindong Wang, Xixu Hu, Wenxin Hou, Hao Chen, Runkai Zheng, Yidong Wang, Linyi Yang, Haojun Huang, Wei Ye, Xiubo Geng, Binxin Jiao, Yue Zhang, and Xing Xie. 2023b. On the Robustness of ChatGPT: An Adversarial and Out-of-distribution Perspective. *ArXiv preprint*, abs/2302.12095.

Peiyi Wang, Lei Li, Liang Chen, Zefan Cai, Dawei Zhu, Binghuai Lin, Yunbo Cao, Qi Liu, Tianyu Liu, and Zhifang Sui. 2023c. Large Language Models are not Fair Evaluators. *ArXiv preprint*, abs/2305.17926.

Wikipedia contributors. 2004. Plagiarism — Wikipedia, the free encyclopedia. [Online; accessed 22-July-2004].

Ledell Wu, Fabio Petroni, Martin Josifoski, Sebastian Riedel, and Luke Zettlemoyer. 2020. Scalable zero-shot entity linking with dense entity retrieval. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6397–6407, Online. Association for Computational Linguistics.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-Pack: Packaged Resources To Advance General Chinese Embedding. *ArXiv preprint*, abs/2309.07597.

Qiming Xie, Zengzhi Wang, Yi Feng, and Rui Xia. 2023. Ask Again, Then Fail: Large Language Models' Vacillations in Judgement. *ArXiv preprint*, abs/2310.02174.

Shuo Yang, Wei-Lin Chiang, Lianmin Zheng, Joseph E. Gonzalez, and Ion Stoica. 2023. Rethinking Benchmark and Contamination for Language Models with Rephrased Samples. *ArXiv preprint*, abs/2311.04850.

Jifan Yu, Xiaozhi Wang, Shangqing Tu, Shulin Cao, Daniel Zhang-Li, Xin Lv, et al. 2023. KoLA: Carefully Benchmarking World Knowledge of Large Language Models. *ArXiv preprint*, abs/2306.09296.

Xinghua Zhang, Bowen Yu, Haiyang Yu, Yangyu Lv, Tingwen Liu, Fei Huang, Hongbo Xu, and Yongbin Li. 2023. Wider and Deeper LLM Networks are Fairer LLM Evaluators. *ArXiv preprint*, abs/2308.01862.

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. *ArXiv preprint*, 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. *ArXiv preprint*, abs/2306.05685.

Kaijie Zhu, Jiaao Chen, Jindong Wang, Neil Zhenqiang Gong, Diyi Yang, and Xing Xie. 2023a. DyVal: Graph-informed Dynamic Evaluation of Large Language Models. *ArXiv preprint*, abs/2309.17167.

Kaijie Zhu, Jindong Wang, Jiaheng Zhou, Zichen Wang, Hao Chen, Yidong Wang, Linyi Yang, Wei Ye, Yue Zhang, Neil Zhenqiang Gong, and Xing Xie. 2023b. PromptBench: Towards Evaluating the Robustness of Large Language Models on Adversarial Prompts. *ArXiv preprint*, abs/2306.04528.

Wenhong Zhu, Hongkun Hao, Zhiwei He, Yunze Song, Yumeng Zhang, Hanxu Hu, Yiran Wei, Rui Wang, and Hongyuan Lu. 2023c. CLEAN-EVAL: Clean Evaluation on Contaminated Large Language Models. *ArXiv preprint*, abs/2311.09154.

A Customized Benchmark Construction based on StructEval

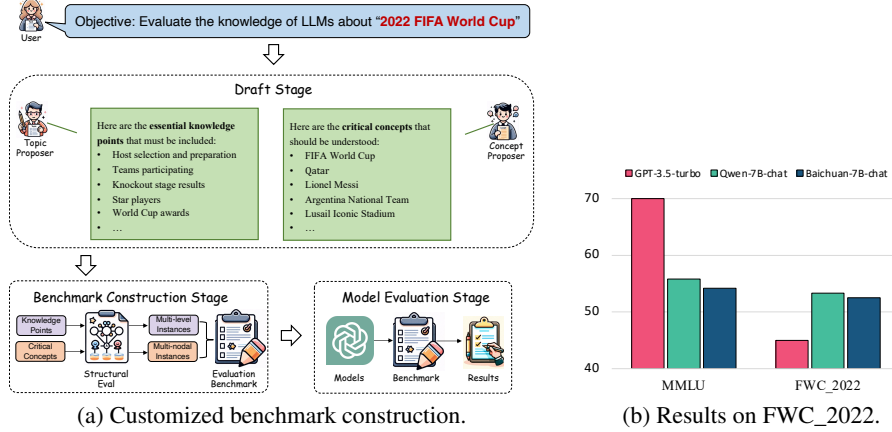(a) Customized benchmark construction.   (b) Results on FWC_2022.

Figure 4: In addition to expanding on existing benchmarks, StructEval can also function as a customized benchmark construction framework. It is capable of automated data construction and evaluation tailored to assessment objectives of any granularity.

The majority of current benchmarks assess models in a static and coarse-grained manner. They typically start by defining a broad assessment domain, such as general knowledge, medical knowledge, or legal knowledge, and then extensively collect questions and answers within that domain. These instances are then fixedly used to evaluate models.

However, with the rapid development of large-scale models, this assessment paradigm faces two issues: 1) The obsolescence rate of static assessments is accelerating, and they are prone to rapid invalidation after reaching benchmark saturation. 2) As the application scenarios of large models become more diverse and refined, such coarse-grained assessment methods struggle to meet the rapidly growing needs for customization in real-world scenarios. For example, evaluating an AI assistant designed to aid in railway museum explanations should target "railway knowledge" rather than "general knowledge". Manually collecting data for each of these customized scenarios is not feasible.

Benefiting from the automatic and dynamic features of StructEval, we can restructure it into a multi-agent-based customized benchmark construction framework. As illustrated in Figure 4a, given a customized assessment objective (e.g., 2022 FIFA World Cup), two agents including topic proposer and concept proposer would list the essential test objectives and important concepts for comprehensive evaluate LLMs within target objective. Then, StructEval would follow the same procedures in Figure 2, and automatically construct a multi-level and multi-nodal benchmarks for evaluation. In order to validate the effectiveness of our approach, we followed the aforementioned steps with GPT-4 to construct a small-scale dataset named FWC_2022, with "2022 FIFA World Cup" as the evaluation objetive. Subsequently, we compare the performance of various models on both a large-scale general benchmark MMLU and FWC_2022. FWC_2022 comprises a total of 240 multiple-choice questions pertaining to various aspects of "2022 FIFA World Cup." Please refer to the appendix for details and instances of the dataset due to page limitations.

The results in Table 4b demonstrate the necessity for customized fine-grained evaluations: 1) In the large-scale general benchmark MMLU, GPT-3.5-turbo perform significantly better than other two LLMs, which indicate that the GPT-3.5 has a stronger general knowledge ability. 2) However, the knowledge cuttoff of GPT-3.5-turbo is September, 2021. Therefore, in FMC_2002, the evaluate datasets about "2022 FIFA World Cup", GPT-3.5-turbo perform worse than other two LLMs which are newly released. The inconsistent conclusion between these two benchmarks indicate that previous static and fine-grained evaluation could not adapt to many scenarios, and StructEval could serve as an valuable tool for a customized, dynamic and fine-grained evaluation automatically.

B Examples of Test Instances

Here is an example of generated instances by StructEval.

Level	Instance	Explanation
Remembering	What is the cranial nerve number associated with the facial nerve? A.CN VII B.CN VI C.CN V D.CN VIII Correct Answer: A.CN VII	As mentioned in the provided document, the facial nerve is also known as the seventh cranial nerve, cranial nerve VII, or simply CN VII.
Understanding	Where does the facial nerve exit the skull? A.Temporal bone B.Parietal bone C.Occipital bone D.Frontal bone Correct Answer: A.Temporal bone	As mentioned in the document, the facial nerve typically exits the skull at the stylomastoid foramen, which is located in the temporal bone.
Applying	Which type of lesion results in a CN VII palsy, manifested as both upper and lower facial weakness on the same side of the lesion? A.Lower motor neuron lesion B.Upper motor neuron lesion C.Trigeminal nerve lesion D.Vestibulocochlear nerve lesion Correct Answer: A.Temporal bone	As per the document, lower motor neuron lesions can result in a CN VII palsy, manifested as both upper and lower facial weakness on the same side of the lesion.
Analyzing	What type of lesion would result in both upper and lower facial weakness on the same side of the face? A.Central seven lesion B.Upper motor neuron lesion C.Lower motor neuron lesion D.Trigeminal nerve lesion Correct Answer: C.Lower motor neuron lesion	As mentioned in the provided document, lower motor neuron lesions can result in a CN VII palsy, manifested as both upper and lower facial weakness on the same side of the lesion.
Evaluating	What is the developmental origin of the facial nerve? A.First pharyngeal arch B. Second pharyngeal arch C.Fifth pharyngeal arch D.Third pharyngeal arch Correct Answer: B. Second pharyngeal arch	According to the document, the facial nerve is developmentally derived from the second pharyngeal arch
Creating	Propose a study to investigate the effectiveness of various taste rehabilitation techniques in patients with facial nerve damage affecting the anterior two-thirds of the tongue. A.Assess the improvement in taste discrimination following acupuncture therapy. B. Compare the efficacy of electronic stimulation versus traditional flavor training C.Evaluate the impact of vitamin supplementation on taste recovery D.Monitor changes in taste sensation after targeted facial massage Correct Answer: B	Given that the facial nerve is involved in the conveyance of taste sensations from the anterior two-thirds of the tongue, comparing electronic stimulation (mimicking natural nerve impulses) with traditional flavor training (using different flavored solutions) directly addresses the rehabilitation of taste function.

Table 6: The generated instances about the test objective "*Facial nerve*" which is sampled from MMLU.

C Framework Design Details

This section will introduce the more details about our framework design.

C.1 Instance Generation based on Bloom's Taxonomy

C.1.1 Test Objective Extraction Instruction

We use the following instruction to identify the underlying test objective for a seed instance.

> # Instruction
> As an expert in education and assessment, your task is to accurately identify the test objective of the seed questions I present, and provide a brief description of that test objective. I will first provide some reference examples. Please ensure that your responses follow a consistent format in line with the provided examples.
> Response in the following format:
> Test Objective: <test objective of the instance>
> Description: <description of the test objective>
>
> # Example 1
> Question: During the third stage of the demographic transition model, which of the following is true?
> A. Birth rates increase and population growth rate is less rapid.
> B. Birth rates decline and population growth rate is less rapid.
> C. Birth rates increase and population growth rate increases.
> D. Birth rates decrease and population growth rate increases.
> Correct Answer: B
> **Test Objective: demographic transition**
> Description: In demography, demographic transition is a phenomenon and theory which refers to the historical shift from high birth rates in societies with minimal technology, to low birth rates in societies with advanced technology.
>
> # Input
> Question: <seed question>
> Options: <question options>
> Correct Answer: <answer>
> Test Objective:

Table 7: Prompt design for test objective extraction.

C.1.2 Details of Instance Selection and Refinement

The post-processing modules are crucial for ensuring the quality of generated instances since there exist several issue for test instances directly generated by LLMs. Table 9 present a case study to demonstrate how this post-processing module to filter out candidate instances with low quality. And Table 8 shows the instruction for retrieval-augmented generation.

> # Instruction
> Refer to the document, select the correct answer for the multiple choice questions about subject.
> If you can find the correct answer in the document, response with the correct choice such as 'A/B/C/D'.
> If you cannot find the correct answer in the document, response with 'cannot answer'
> If the choices contain more than one correct option, esponse with 'cannot answer'
> Ensure your response begin with the correct choice and do not output any other content.
>
> # Input
> Document: <Support document>
> Question: <Generated question>
> Choices: <Options>

Table 8: Prompt design for RAG.

C.2 Instance Expansion based on Concept Mapping

C.2.1 Critical Concepts Extraction

We use the following instruction to extract the critical concepts that must be understood to correctly answer the seed question.

Issue	Example	Solution
Context-dependent	According to the provided document, what was the impact of Comet Shoemaker–Levy 9 in July 1994?	Prompt LLM to remove
Incorrect answer	What is the principle used in satellite navigation to determine exact distances? Generated answer: Time synchronization	Use RAG to filter
Multiple answers	What is the most important fish species produced in fish farming worldwide? A.Tuna B.Carp C.Mussels D.Salmon	Use RAG to filter
Cannot verify	To transform the mood of 'Seasons' into a more introspective and melancholic tone, what instrumental addition would be most appropriate for the song? (cannot verify answer correctness based on the document)	Use RAG to filter
Too easy instance	What is the most commonly used approach for length measurement? Generated answer: rulers	Construct a model pool to filer

Table 9: Issues exist in directly generated candidate instances by LLM. And the corresponding solutions for filtering the instances with low quality. The examples are sampled from the benchmark construct process of StructEval.

 # Instruction
As an expert in education and assessment, your task is to identify the key concepts and their related knowledge that must be understood in order to answer a given seed question. For each seed question, list all important concepts and provide a brief description for each concept. I will first provide some reference examples. Please ensure that your responses follow a consistent format in line with the provided examples.
Response in the following format, each line include an concept:
{'name': <concept name>, 'description': <concept description>}

 # Example 1
Question: During the third stage of the demographic transition model, which of the following is true?
A. Birth rates increase and population growth rate is less rapid.
B. Birth rates decline and population growth rate is less rapid.
C. Birth rates increase and population growth rate increases.
D. Birth rates decrease and population growth rate increases.
Correct Answer: B
Critical Concepts:
{'name': 'third stage of demographic transition', 'description': 'In stage three of demographic transition...'}
{'name': 'birth rates', 'description': 'Birth rate is the total number of live human births per 1,000 population...'}
{'name': 'population growth rate', 'description': 'Population growth is the increase in the number of people...'}

 # Input
Question: <seed question>
Options: <question options>
Correct Answer: <answer>
Critical Concepts:

Table 10: Prompt design for critical concepts extraction.

C.2.2 Helpful Knowledge Triplets Selections

We use the following instruction to select helpful knowledge triplets from all candidates.

Instruction
Identify up to three fact triples that are most helpful to comprehend the provided question. Refer to the provided examples for valid response. If none of them is helpful, output [None].

Example 1
<Start of Question>
Question: Which of the following best describes the structure that collects urine in the body?
A. Bladder B. Kidney C. Ureter D. Urethra
<End of Question>
Candidate Triplets:
1. (urine, subclass of, secretion or excretion)
2. (urine, UMLS CUI, C2963137)
3. (urinary bladder, connects with, urethra)
4. (urinary bladder, part of, urinary system)
5. (urinary bladder, subclass of, particular anatomical entity)
Relevant Triplets:
3. (urinary bladder, connects with, urethra)
4. (urinary bladder, part of, urinary system)

Input
Question: <seed question>
Candidate Triplets: <candidate triplets>
Relevant Triplets:

Table 11: Prompt design for helpful knowledge triplets selection.

D Human Annotation Guidelines

Here is the annotation guidelines for our human evaluation is shown in Figure 5. We recruit 5 annotators to participate in the human evaluation, each of whom possesses a bachelor degree. To ensure the clarity and consistency in the evaluation, we provided detailed instructions and examples in the annotation guidelines. Each instance is annotated by 3 participants, and the final results are determined by a majority vote.

E Error Analysis of Constructed Benchmark

As we discussed in Section 4.1, according to the human evaluation results, there still exist a fewer instances which not meet the standard. In order to find the underlying causes of these errors, we conduct a detailed error analysis which is demonstrated in Table 12, 13, and 14.

Annotation Guidelines for StructEval

Please strictly follow the guidelines, and evaluate the quality of the given test instance from three perspectives including **Instance Helpfulness, Question Answerability, and Answer Correctness**.

A test instance is labeled as unhelpful if:

- **The extracted test objective does not match the seed instance.**
  e.g., Seed question: which one is not a variety of apple? Test Objective: product of apple inc
- **The generated instance does not conform to the test objective.**
  e.g., Test objective: apple cultivars. Generated question: What are the benefits of eating apples?
- **The generated instance does not conform to the corresponding concept.**
  e.g., Concept: Fuji apple. Generated question: Where is mountain Fuji located?

A test instance is labeled as unanswerable if:

- **The generated instance requires external document to answer.**
  e.g., Which is the highest mountain mentioned in the provided document?
- **The generated instance contains multiple correct choices**
  e.g., Which one is eatable? A.Apple B.Orange, C.Banana D.Wood
- **The generated instance lacks of necessary information for answering the question.**
  e.g., Which is his favorite movie?

A test instance is labeled as incorrect if:

- **The generated answer is incorrect to the question.**
  e.g., Question: Which country is the 2022 World Cup champion? Generated answer: France

---

Here is the data to be evaluated:

Test objective: Facial nerve

The corresponding document: Voluntary facial movements, such as wrinkling the brow, showing teeth, frowning, closing the eyes tightly (inability to do so is called lagophthalmos), pursing the lips and puffing out the cheeks, all test the facial nerve. There should be no noticeable asymmetry. In an upper motor neuron lesion, called central seven, only the lower part of the face on the contralateral side will be affected, due to the bilateral control to the upper facial muscles (frontalis and orbicularis oculi). Lower motor neuron lesions can result in a CN VII palsy (Bell's palsy is the idiopathic form of facial nerve palsy), manifested as both upper and lower facial weakness on the same side of the lesion....

Question:What is the developmental origin of the facial nerve? A.First pharyngeal arch B.Third pharyngeal arch C.Fourth pharyngeal arch D.Second pharyngeal arch

Answer: D.Second pharyngeal arch
Explaination: As mentioned in the document, the facial nerve is developmentally derived from the second pharyngeal arch, or branchial arch.

| Helpful | Unhelpful | | Answerable | Unanswerable | | Correct | Incorrect |

| Submit |

Figure 5: The annotation guidelines for our human evaluation.

Test Objective	Example	Cause
	Answerability Issue	
Economic growth	Question: What constraints to economic growth are highlighted? A. Government intervention B. Depleted resources and energy consumption C. Technological advancements D. Increased labor force	Context-dependent instance not be filtered
Moon	What will happen in 50 billion years according to the text? A. The Moon's rotation will stop B. The Sun will become a red giant C. The Moon will collide with Earth D. The Earth's rotation will match the Moon's orbital period	Context-dependent instance not be filtered
Dementia	What do physicians need to include in any dementia evaluation? A.A memory assessment B.A somatic disturbance evaluation C. A depression screening D.A sensory function test	Instance with multi-answers not be filtered by RAG
Fish farming	What is the most important fish species produced in fish farming worldwide? A.Tuna B.Carp C.Mussels D.Salmon	Instance with multi-answers not be filtered by RAG
Human subject research	Question: What are the three guidelines that serve as the baseline for the report? A. Beneficence, justice, and respect for persons B. Beneficence, integrity, and respect C. Prudence, integrity, and honesty D. Beneficence, justice, and honesty	Unclear reference

Table 12: Error analysis of answerability issues for benchmark constructed by StructEval.

Test Objective	Example	Cause
	Helpfulness Issue	
Premises Liability	In 'peralta v. henriquez', what was the specific dangerous condition that led to the accident? A.Lack of illumination B.Lack of security personnel C.Poor maintenance of the premises D.Inadequate signage	Unhelpful for test objective assessment
psychology	Who is considered the first director of Harvard's psychological laboratory and a student of Wilhelm Wundt? A. Scott Lilienfeld B. Saul Kassin C. Thomas Bond D. Hugo Münsterberg	Unhelpful for test objective assessment
GDP calculation	How often does India change the base year for its GDP calculation, according to the Frontier Strategy Group? A. Every 10 years B. Every 3 years C. Every 5 years D. Every 7 years	Unhelpful for test objective assessment

Table 13: Error analysis of Helpfulness issues for benchmark constructed by StructEval.

Test Objective	Example	Cause
	Correctness Issue	
States' rights	What was the outcome of the U.S. Supreme Court's decision in the case of California Proposition 14, and what did this decision overturn? A. The decision overturned Proposition 14, allowing discrimination in housing. B. The decision upheld the Rumsford Fair Housing Act, allowing discrimination in housing. C. The decision upheld Proposition 14, banning discrimination in housing. D. The decision overturned the Rumsford Fair Housing Act, banning discrimination in housing.	Conflict with document
Solar System	How do the inner planets in the Solar System differ from the inferior planets? A. The inner planets are closer to the Sun than the inferior planets. B. The inner planets are larger in size compared to the inferior planets. C. The inner planets have atmospheres substantial enough to generate weather, while the inferior planets do not. D. The inner planets are composed mainly of gases, while the inferior planets are composed mostly of rocky materials.	Incorrect answer
Physical Weathering	What type of physical weathering is considered the most important? A. Thermal fracturing B. Frost wedging C. Pressure release D. Wedging by plant roots	Incorrect answer

Table 14: Error analysis of Correctness issues for benchmark constructed by StructEval.