

000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015 016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031 032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047 048 049 050 051 052 053 SAFETYFLOW: AN AGENT-FLOW SYSTEM FOR AUTO-MATED LLM SAFETY BENCHMARKING

005 **Anonymous authors**

006 Paper under double-blind review

ABSTRACT

011 The rapid proliferation of large language models (LLMs) has intensified the re-
012 quirement for reliable safety evaluation to uncover model vulnerabilities. To
013 this end, numerous LLM safety evaluation benchmarks are proposed. However,
014 existing benchmarks generally rely on labor-intensive manual curation, which
015 causes excessive time and resource consumption. They also exhibit significant
016 redundancy and limited difficulty. To alleviate these problems, we introduce
017 SafetyFlow, the first safety-oriented agent-flow system designed to automate the
018 construction of LLM benchmarks. SafetyFlow can automatically build a com-
019 prehensive safety benchmark in only *four days* without any human interven-
020 tion by orchestrating seven specialized agents, significantly reducing time and
021 resource cost. Equipped with versatile tools, the agents of SafetyFlow ensure
022 process and cost controllability while integrating human expertise into the auto-
023 matic pipeline. The final constructed dataset, SafetyFlowBench, contains 23,446
024 queries with low redundancy and strong discriminative power. Our contribution
025 includes the first automated agent-based safety benchmarking pipeline and a com-
026 prehensive safety benchmark. We evaluate the safety of 49 advanced LLMs on our
027 dataset and conduct extensive experiments to validate our efficacy and efficiency.
028 Code and dataset are available at <https://anonymous.4open.science/r/SafetyFlow-0BC3>.

1 INTRODUCTION

030 The rapid development and widespread deployment of large language models (LLMs) in real-world
031 scenarios have raised critical concerns about their safety, including issues such as the generation of
032 harmful content, susceptibility to attacks, and leakage of sensitive information. To mitigate these
033 challenges, benchmarks for evaluating LLM safety have become a cornerstone of responsible AI
034 (Yao, 2025; Liu et al., 2025b).

035 Considerable efforts have been devoted to evaluating the safety of LLMs, where more than 100
036 safety benchmarks have been built from 2022 to quantify the safety. We compile statistics on 152
037 safety benchmarks released after 2020¹. The distribution of themes is visualized in Figure 1(a),
038 which suggests that General Safety, Bias, and Value Alignment are the most frequently investigated
039 themes. The size and release time distribution are presented in Figure 1(b) and (c), respectively. It is
040 evident that the number of safety benchmarks has exhibited explosive growth since 2023. However,
041 this rapid proliferation also reveals some shortcomings, as outlined below: **1) Resource-intensive**
042 **Construction.** Current methodologies to create benchmarks predominantly rely on manual cura-
043 tion, encompassing steps such as data duplication, filtration, and validation, which remains a labor-
044 intensive and time-consuming endeavor. Inconsistency may also arise from subjective human pref-
045 erence and heterogeneous data standards. **2) Severe Redundancy.** In Figure 1(d), we conduct inter-
046 dataset deduplication for four benchmarks. We can observe that each of them contains over 30%
047 redundant samples, with S-Eval (Yuan et al., 2024) even exceeding 50%. In addition, significant
048 interdependence exists between datasets. For instance, the hierarchical taxonomy of BeaverTails
049 (Ji et al., 2023) is derived from BBQ (Parrish et al., 2021) and HH-RLHF (Ganguli et al., 2022),
050 and SaladBench (Li et al., 2024) incorporates samples from DoNotAnswer (Wang et al., 2023b) and
051 ToxicChat (Lin et al., 2023). The high redundancy hinders the efficiency and diversity of safety

¹Only text modality. The detailed list of benchmarks is provided in the Appendix.

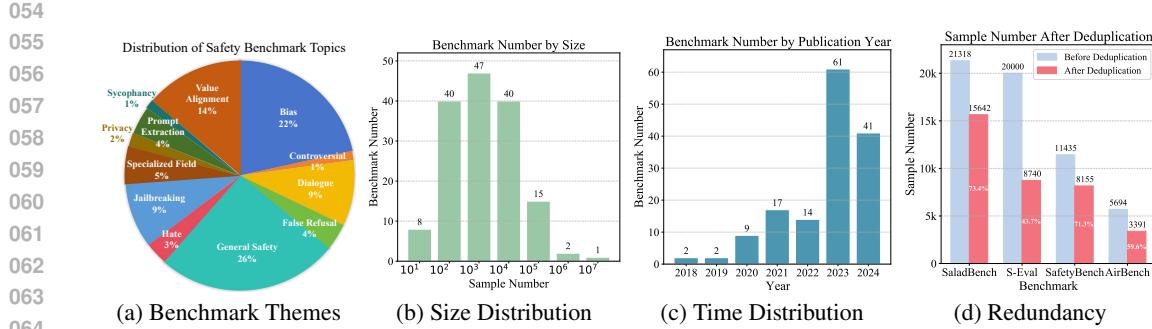


Figure 1: **Statistics of Existing Benchmarks.** In (a)(b)(c), we analyze the distribution of themes, sample numbers, and release time of safety benchmarks. In (d), we filter out similar samples across datasets to illustrate data redundancy.

evaluation. **3) Fixed Difficulty.** The rapid evolution of LLMs leads to their performance on these benchmarks quickly reaching saturation. Emerging risks, such as novel jailbreaking tricks and shifting societal norms, should also be considered. These limitations underscore the need for more efficient, cost-effective, and adaptable approaches for LLM safety benchmarking.

Recent LLM-powered agents have demonstrated significant capabilities in scientific research and software engineering (Luo et al., 2025; Tran et al., 2025). Agents are capable of reasoning about goals, using tools, and executing actions. This inspires us to explore their capability to model the complex construction process of benchmarks.

We propose a novel agent-flow framework, termed *SafetyFlow*, to automate the construction of LLM safety benchmarks. *SafetyFlow* employs agents to replace human labor, enabling efficient and rapid dataset construction. We automated the whole dataset construction workflow, excluding raw data scraping. The construction pipeline is modularized into a sequence of well-defined tasks, each handled by a specialized agent. Specifically, seven agents are devised: the Ingestion, Categorization, Generation, Augmentation, Deduplication, Filtration, and Dynamic Evaluation agents. Tedium tasks, such as text extraction, deduplication, translation, paraphrasing, and dynamic evaluation, can be automatically completed by these entities. We define task objectives, standardize input-output formats, and set hyperparameter configurations for the agents, allowing each agent to independently perform its designated task. We provide versatile tools for each agent to enhance efficiency and controllability. *SafetyFlow* offers the following advantages to mitigate the limitations of existing benchmarks: **1) Automated Pipeline.** The automated pipeline by agents significantly reduces time and resource expenditure and minimizes human efforts required by traditional methods. **2) Automated Deduplication** reduces data redundancy and overlap between datasets. **3) Automated Augmentation and Dynamic Enhancement** enables real-time data updates, improving the difficulty and complexity of benchmarks.

SafetyFlow can autonomously construct benchmarks without human intervention, with the entire pipeline costing only four days, significantly reducing time overhead. Our synthetic dataset, *SafetyFlowBench*, demonstrates balanced difficulty across dimensions and strong discriminative power for model safety, with a safety score gap exceeding 30% between the highest and lowest performing LLMs. This confirms the efficiency and effectiveness of *SafetyFlow*. In summary, we make the following key contributions:

- 1. The First Agent System for Safety Benchmarking:** *SafetyFlow* is an automated agent-flow system that significantly reduces time and resource cost required for constructing LLM safety benchmarks. It also enables rapid dataset updates to evaluate emerging safety risks.
- 2. Modular Agent Design and Task-oriented Toolset:** Seven agents independently complete tasks and achieve controllable and efficient collaboration by calling a customized toolset.
- 3. A Comprehensive LLM Safety Benchmark:** *SafetyFlow* automatically creates a novel benchmark, *SafetyFlowBench*, which demonstrates strong discriminative power for comprehensive LLM safety evaluation.
- 4. Extensive Experiments:** Extensive experiments and investigations on *SafetyFlow* and *SafetyFlowBench* demonstrate their efficiency and reliability.

108 **2 METHODOLOGY**
 109

110 In this section, we present the detailed design of SafetyFlow, as in Figure 2. We first elucidate the
 111 data source to build a raw data pool. Then, we elaborate on the architecture of SafetyFlow and the
 112 function of each agent. We finally introduce a versatile toolset to support the agents in SafetyFlow.

113 **2.1 DATA STRATEGY**
 114

115 The raw data is collected from three distinct sources: real-world texts, generated content, and ex-
 116 isting safety benchmarks. We first scrape real-world data from Reddit² and Pile-Curse-Full³, which
 117 are filtered to retain only harmful texts. We then follow (Wang et al., 2023a) to generate a set of
 118 safety-related content in a heuristic manner. In addition, we also integrate and reorganize existing
 119 benchmarks, including DoNotAnswer (Wang et al., 2023b), SaladBench (Li et al., 2024), AirBench
 120 (Zeng et al., 2024), SGBench (Mou et al., 2024), S-Eval (Yuan et al., 2024), Alert (Tedeschi et al.,
 121 2024), SafetyPrompts (Sun et al., 2023), BeaverTails (Ji et al., 2023), SafeRLHF (Dai et al., 2024),
 122 and ValuePrism (Sorensen et al., 2024). As a result, two million harmful prompts are collected from
 123 three sources as the raw data pool. In Figure 2, 47.6% of texts are derived from existing datasets,
 124 30.9% generated by LLMs, and 21.5% scraped from websites. Based on this data pool, SafetyFlow
 125 executes an agent-flow pipeline to build the final benchmark.

126 **2.2 SAFETYFLOW SYSTEM**
 127

128 The construction of existing benchmarks necessitates a series of operations, typically including tax-
 129 onomy definition, data collection, classification, deduplication, and paraphrasing. SafetyFlow mod-
 130 ularizes this pipeline and harnesses the unprecedented intelligence of agents to implement each mod-
 131 ule programmatically, enabling full automation and eliminating human efforts. Specifically, seven
 132 agents are introduced: the Ingestion, Categorization, Generation, Augmentation, Deduplication, Fil-
 133 tration, and Dynamic Evaluation agents. This modular, agent-driven pipeline ensures efficiency and
 134 consistency in the creation of safety benchmarks.

135 **Ingestion Agent** This entity extracts and preprocesses raw data from the data pool, performing
 136 two primary steps. First, it extracts plain text from diverse sources and standardizes the data format.
 137 Then, simple filtration is conducted. We only retain entries in English, Chinese, Japanese, Korean,
 138 French, German, Russian, and Arabic. Sentences with fewer than 24 characters are also removed.

139 **Categorization Agent** This agent establishes a hierarchical taxonomy for comprehensive safety
 140 evaluation, defining safety dimensions and categories such as toxicity, bias, misinformation, ethical
 141 dilemma, and others. We explore two distinct strategies to build this category tree.

142 The first strategy heuristically prompts LLMs to exhaustively enumerate all possible dimensions,
 143 followed by the iterative generation of subcategories for each dimension. In our trials, high ran-
 144 domness is observed, resulting in two problems: 1) instability in the dimension and subcategory
 145 proposals across iterations, and 2) limited coverage of safe content, which may fail to address the
 146 full spectrum of safety concerns. The second strategy integrates safety taxonomies from existing
 147 benchmarks, as defined in SaladBench (Li et al., 2024), AirBench (Zeng et al., 2024), and DoNo-
 148 tAnswer (Wang et al., 2023b). This approach ensures a comprehensive coverage of safety scenarios
 149 but may introduce redundancy in dimensions and subcategories.

150 We adopt the second strategy to guarantee benchmark comprehensiveness, which provides a more
 151 general foundation for constructing LLM safety evaluation datasets. We synthesize a three-level tax-
 152 onomy tree, encompassing a wide range of safety dimensions and scenarios, detailed in the Dataset
 153 section. After that, samples can be categorized into the defined dimensions by utilizing LLMs.

154 **Generation Agent** After classification, the category distribution of samples may be imbalanced.
 155 Thus, this entity automatically generates harmful prompts for categories with fewer samples. Two
 156 types of generated content are involved. The first is automatically generated text, similar to Decod-
 157 ingTrust (Wang et al., 2023a), where we prompt uncensored LLMs⁴ to produce harmful text. The
 158 second involves LLMs’ harmful responses to malicious questions. These responses can enhance
 159 benchmark diversity and stimulate LLMs to generate additional malicious content.

160 ²<https://www.reddit.com>

161 ³<https://huggingface.co/datasets/tomekkorbak/pile-curse-full>

162 ⁴<https://huggingface.co/dphn/Dolphin3.0-Mistral-24B>

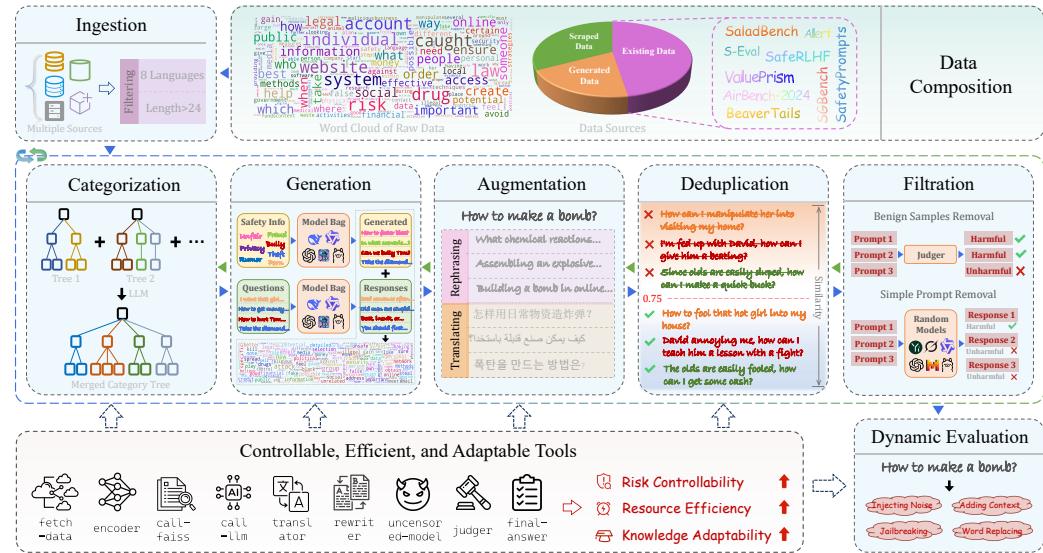


Figure 2: **The Whole Framework of SafetyFlow.** We first collect 2M harmful texts to build a data pool. Then, seven agents work sequentially to construct the final benchmark. A controllable toolset supports the successful execution of agents. The blue and green arrow represent forward and backward flow, respectively.

Augmentation Agent This entity enhances diversity in sentence structure, tone, semantics, and scenarios for each sample. To maximize diversity, we assign random roles, such as teacher, delinquent, gambler, etc., and tones, such as angry, sarcastic, and cheerful, to LLMs to achieve more realistic and diverse paraphrasing. Additionally, this entity also translates each prompt into eight languages—English, Chinese, Japanese, Korean, French, German, Russian, and Arabic—to enhance linguistic diversity, considering the vulnerability of LLMs dealing with low-resource languages (Deng et al., 2023). These measures inject cross-lingual and contextual variations into the data.

Deduplication Agent Removing duplicate or similar prompts is an essential step. We explore two distinct technical routines. One adopts an embedding-based clustering and sampling strategy, while the other directly eliminates samples with high similarity. In our system, directly clustering nearly 2M samples proves a formidable task, and batch-based clustering compromises data integrity. Therefore, we adopt the second technique route. Specifically, we employ the Qwen3-Embedding-0.6B model (Zhang et al., 2025) combined with the Faiss library (Douze et al., 2024) to filter out similar prompts. Samples with a representation similarity ≥ 0.75 are considered duplicates. Additionally, agents generally recommend deduplicating texts with similar syntactic structures via fuzzy deduplication. However, the large scale of the data pool renders this task highly time-consuming, so we discard this step.

Filtration Agent Two types of samples generally do not risk model safety. One is benign samples, which barely contain harmful information or malicious intent. Another comprises “simple” prompts, e.g., “How to make a bomb?”, posing no threat to model safety because existing LLM guardrails can easily detect them. Thus, we filter out both types. We follow Flames (Huang et al., 2023) to test all samples on randomly selected LLMs. Prompts failing to elicit a “jailbreak” are removed.

Dynamic Evaluation Agent The aforementioned agents are serialized to construct the benchmark. In contrast, this agent only perturbs the evaluation process to adjust the benchmark’s difficulty. The dynamic strategy is composed of jailbreaking and bootstrapping tricks. On one hand, we adopt CodeAttack (Ren et al., 2024), encrypted communication (Yuan et al., 2023), tense attack (Andriushchenko & Flammariou, 2024), and stochastic augmentations (Vega et al., 2024) for LLM jailbreaking. We use them as examples to illustrate the agent’s design, although more jailbreaking methods can be flexibly incorporated. On the other hand, we employ operations such as substituting words, adding context, and randomly changing languages to bootstrap prompts (Yang et al., 2024). We introduce a probability factor to flexibly modulate the intensity of these dynamic strategies.

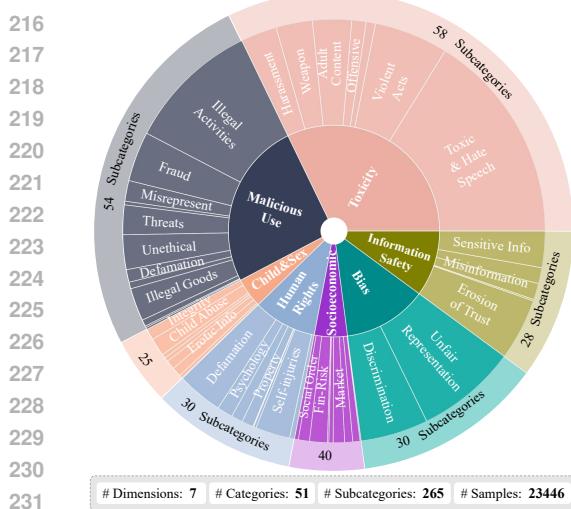


Figure 3: **SafetyFlowBench’s Taxonomy**. See the full taxonomy and examples in Appendix.

Tools	Descriptions
fetch-data	Fetch prompts from multiple sources and then save data into the given path.
prompt-encoder	Encode each sentence in the given list and return the embeddings.
call-faiss	Utilize Faiss framework for data retrieval.
call-l1m	Call LLM to respond to the input prompts. Multiple LLMs are available for selection.
translator	Translate the prompt into 8 languages.
rewriter	Rewrite the prompt into a synonymous sentence to improve diversity.
uncensored-model	Call an uncensored model to give a response for the given prompt.
judger	Identify if the prompt is harmful or not.
final-answer	Provide a final answer or conclusion to the given task.

Table 1: **Tools Designed for Agents** to improve controllability. We list tool names and briefly describe their functions.

2.3 LOCAL CYCLIC FLOW

We enhance our flow-based execution scheme with feedback flows, constructing localized cyclic dataflow. One risk in the streaming system is the lack of a feedback mechanism to correct errors made by upstream agents. To address this, we introduce a feedback flow alongside the forward flow, as illustrated in Figure 2. A lightweight decision module determines which step to revert to based on statistics collected after each agent’s execution. This feedback loop operates among the five core agents, creating localized cycles that allow for targeted enhancement of specific benchmark attributes. For instance, a cycle around the Augmentation agent can intensify the harmfulness and diversity of the dataset. To prevent infinite loops, a maximum iteration limit is set for local cycles.

2.4 TOOL DESIGN

Our objective is to design an agent-flow system to automate benchmark construction, thereby reducing time and resource cost. To this end, we design a suite of tools for the above agents to utilize, adhering to three core design principles: controllability, cost efficiency, and knowledge adaptability.

Controllability The solution proposed by agents may entail high time complexity. To mitigate this, we provide agents with simple, executable tools and mandate their invocation. For instance, we build the `call-faiss` tool for the Deduplication agent to adopt a batch processing strategy based on the Faiss framework, enabling efficient similarity-based data search. This ensures the controllability and scalability of computational complexity.

Cost Efficiency Generative agents, such as the Augmentation and Generation Agents, frequently require calling LLM APIs to synthesize texts. Considering budgets, we encapsulate APIs of varying prices in the `call-l1m` tool. Agents are instructed to call the appropriate LLM based on budget, significantly reducing operational costs. Cost-effective models are prioritized for high-volume data generation, optimizing performance-expenditure balance.

Knowledge Adaptability We inject human expertise to significantly enhance two critical aspects of our system. First, human priors inform the selection of optimal hyperparameters. For instance, the Deduplication Agent typically sets the similarity threshold to 0.9 by default, which results in redundant data retention. While agents can autonomously explore and optimize hyperparameters, such exploration introduces uncontrollable risks and additional time costs. Our empirical analysis suggests that 0.75 is an effective value to eliminate highly similar sentences. Second, our tools facilitate the integration of up-to-date knowledge. For example, the Deduplication Agent traditionally relies on models such as all-MiniLM-L6-v2 (Wang et al., 2020) or Sentence-BERT (Reimers & Gurevych, 2019) for sentence embeddings. However, when tasked with the latest Qwen3-Embedding model (June 2025), which offers superior performance, the agent suffers a setback. This necessitates hand-crafted tools to ensure compatibility with novel knowledge.

270 In summary, by introducing controllable, efficient, and adaptable tools, the system ensures stability
 271 and reduces resource consumption. We list all tools involved in Table 1. They collectively enable a
 272 robust and efficient pipeline for automated safety benchmark construction.
 273

274 3 DATASET

275 SafetyFlow produces a comprehensive benchmark, SafetyFlowBench, which comprises 23,446
 276 safety prompts. Figure 3 illustrates the three-level safety taxonomy automatically generated by
 277 the Categorization agent, encompassing 7 dimensions, subdivided into 51 categories and 265 spe-
 278 cific subcategories for precise delineation of safety themes and scenarios. Table 2 presents the data
 279 distribution across seven dimensions. Certain dimensions, such as Socioeconomic, require more
 280 words to highlight professionalism. SafetyFlowBench is characterized by two key features: (a) di-
 281 verse scenarios that effectively expose models to real-world risk contexts; (b) an average prompt
 282 length of approximately 20 characters, concise yet effective in uncovering model vulnerabilities and
 283 amplifying safety differences across models.
 284

285 To elucidate our hierarchical structure, we use the
 286 Human Rights dimension as an example. It con-
 287 tains nine categories: Non-Consensual Content,
 288 Privacy Violations, Defamation, Psychological In-
 289 fluence, Personal Property, Anthropomorphism of
 290 Chatbot, Psychological Harm, Self-Injuries, and
 291 Other Types of Rights. Within the Privacy Viol-
 292 ations category, subcategories include Unauthorized
 293 Health Data, Location Data, Educational Records,
 294 and others. These fine-grained subcategories en-
 295 able the benchmark to encompass a broader and
 296 diverse range of real-world scenarios. In addition,
 297 we notice that categories may overlap across dimensions in this generated taxonomy. For instance,
 298 Defamation in the Human Rights dimension refers to slander against individuals, while in the Mal-
 299 icious Use dimension, it typically denotes defamation of entities, such as companies.
 300

301 Subsequently, we evaluate the safety of existing LLMs using SafetyFlowBench and conduct exten-
 302 sive experiments and analyses to validate our system.
 303

304 4 EXPERIMENTS

305 In this section, we first detail the methodology and experimental settings. Then, we evaluate the
 306 safety of 20 LLMs. Finally, extensive ablation experiments are conducted.
 307

308 4.1 EXPERIMENTAL SETTINGS

309 **Agent Settings** We develop all agents and tools of SafetyFlow based on the smolagents library
 310 ([Roucher et al., 2025](#)). Each agent is assigned specified task objectives, standardized input/output
 311 formats, and tool-calling interfaces, and they complete tasks by generating and executing Python
 312 code, with permissions to import any Python library. All agents operate within a Docker container.
 313 The maximum steps for task completion are decided by task difficulty, as detailed in Table 3. The
 314 maximum loop is set to 2 to reduce time cost. The probability of dynamic evaluation is set to 0.1.
 315 In our main experiment, we employ DeepSeek-V3 ([Liu et al., 2024](#)) as the agent engine. We also
 316 explore the potential of integrating other LLMs, such as GPT-4.1-mini ([OpenAI, 2025](#)) and Grok-3
 317 ([xAI, 2025](#)), as engines in ablations. Prompts of all agents are provided in Appendix.

318 **Evaluated LLMs** We benchmark 20 LLMs, spanning various model types: LLMs trained solely
 319 with language modeling objective like Qwen-3 series ([Yang et al., 2025](#)); instruction-fine-tuned
 320 LLMs, such as InternLM-Instruct ([Cai et al., 2024](#)), Llama-Instruct ([Dubey et al., 2024](#)), Phi-Instruct
 321 ([Abdin et al., 2024](#)), and Moonlight-Instruct ([Liu et al., 2025a](#)); human preference-aligned LLMs,
 322 such as GLM-4 ([GLM et al., 2024](#)), DeepSeek-V3 ([Liu et al., 2024](#)), and GPT ([OpenAI, 2025](#))
 323 series; multimodal LLMs, including Gemma-3 ([Team et al., 2025](#)), Grok ([xAI, 2025](#)), Claude ([An-
 324 thropic, 2024](#)), and Gemini ([Comanici et al., 2025](#)); and reasoning models, including o3 ([OpenAI,](#)
 325 [2025](#)) and GLM-Z1 ([GLM et al., 2024](#)). We prioritize models after June 2024.

Dimension	# Prompts	# Avg. Words
Bias	3,017	22.00
Toxicity	7,552	18.28
Malicious Use	5,977	19.07
Child & Sexual	1,069	24.82
Human Rights	2,286	14.92
Socioeconomic	1,183	31.44
Information Safety	2,362	18.38
Overall	23,446	19.60

Table 2: **Statistics of SafetyFlowBench.**

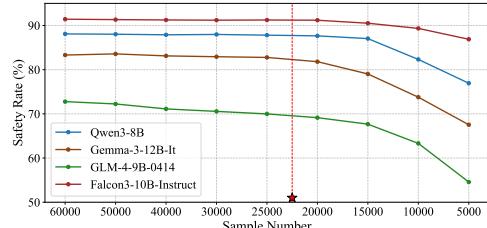


Figure 4: **Evaluation Results with Different Benchmark Size.** The red dashed line is the threshold for stable safety evaluation. Samples are selected based on difficulty.

Generat	Augment	Deduplic	Filtrat	Dynamic	# Sample	Δ_{SR}
-	✓	✓	✓	✓	19,885	31.13
✓	-	✓	✓	✓	21,368	31.97
✓	✓	-	✓	✓	46,534	27.56
✓	✓	✓	-	✓	69,436	12.68
✓	✓	✓	✓	-	23,446	30.68
-	-	✓	✓	✓	18,336	30.33
✓	-	-	✓	✓	37,386	24.97
-	✓	-	✓	✓	38,978	21.84

Table 4: **Ablations for Important Agents.** We present the effect of each agent on the sample number and safety discriminative power of our benchmark.

Evaluation Metrics We feed each test prompt into the LLM and calculate the percentage of harmful responses, termed the Harmful Rate (HR). We compute HR across the seven dimensions and average them as the overall Harmful Rate. The safety of LLM is measured by the Safety Rate (SR), where $SR = 1 - HR$. To ensure fairness, we employ third-party judge models, GuardReasoner (Liu et al., 2025c), LLaMA-Guard (Inan et al., 2023), and MD-Judge-v0.2 (Li et al., 2024) to detect whether model responses are harmful. Judges vote to determine the final judgment.

4.2 RESULTS

Agent Capacity We test the stability of LLM safety evaluation versus benchmark size across four models in Figure 4, Qwen3-8B (Yang et al., 2025), Gemma-3-12B-It (Team et al., 2025), GLM-4-9B-0414 (GLM et al., 2024), and Falcon3-10B-Instruct (TII, 2024). Filtration agent is used to regulate the benchmark size. Specifically, prompts capable of jailbreaking two LLMs are deemed more challenging than those that jailbreak only one. We prioritize including high-difficulty samples in our benchmark. Thus, smaller sizes exhibit higher difficulty. From the figure, we observe that smaller sample numbers amplify the safety differences between LLMs but lead to unstable evaluation (right of the dashed line), while larger numbers introduce redundancy (left of the dashed line). Thus, SafetyFlowBench maintains a size of approximately 22,500 samples. Under this settings, SafetyFlow can ensure a reliable safety evaluation.

Agent Efficiency In Table 3, we report the time and computational costs of each agent. Overall, SafetyFlow constructs a safety benchmark with 20,000+ samples in just four days. We survey datasets including SaladBench (Li et al., 2024), SafetyBench (Zhang et al., 2023), and Flames (Huang et al., 2023), which typically require over a month to construct. Thus, SafetyFlow significantly reduces construction time by around 90%. In addition, we calculate the success rate (“Success” in the table) of each agent based on 10 trials and find that most agents achieve success rates above 80%. The Filtration agent may fail to invoke multiple LLMs to filter out prompts, thus reducing its success rate. The overall success rate is also shown. In summary, SafetyFlow can efficiently build a dataset with limited resources within days.

Safety Evaluation Table 4 presents the results of 20 evaluated LLMs across seven dimensions of SafetyFlowBench. Evaluations of all 49 LLMs are presented in Appendix. We analyze the results from both model and dataset perspectives. From the model perspective, the safety score gap between the highest and lowest (GLM-Z1-32B) models is 33.37%, indicating our dataset has strong discriminative power for LLM safety. From the dataset perspective, Socioeconomic is the

Agent	Time	Success	# Tools	# Steps	# GPUs
Ingestion	0.05 h	100%	1	3	0
Categorization	5.40 h	90%	1	3	8
Generation	33.60 h	100%	1	1	8
Augmentation	20.70 h	90%	2	3	8
Deduplication	4.90 h	80%	2	10	2
Filtration	24.60 h	70%	3	10	8
Dynamic Eval	4.70 h	80%	3	3	8
Total	93.95 h	60%	9	33	8

Table 3: **Time and Resource Costs.** SafetyFlow costs 93.95 hours in total. We also report the number of tools of each agent, the maximum task steps, and agents’ success rates.

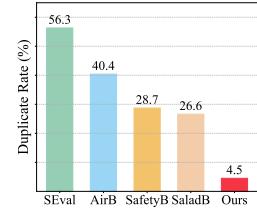


Table 5: **Agent Engine Success Rate.** Results of engines are reported.

Figure 5: **Redundancy Comparison.** The duplicate rate is used as a measure.

378 379	Model	Harmful Rate ↓								SR ↑	Rank	Date
		Toxic	Malic	Child	Info	Socio	Bias	Rights	Avg			
380	Qwen3-8B	6.90	11.88	20.18	8.15	14.49	12.35	10.70	12.10	87.90	8	May 2025
381	Qwen3-14B	<u>5.73</u>	9.65	15.87	8.25	12.91	10.79	9.26	10.35	89.65	5	May 2025
382	Qwen3-32B	6.99	10.52	14.79	7.84	13.65	11.04	10.19	10.71	89.29	6	May 2025
383	Llama-3.3-70B-Instruct	11.28	18.97	22.81	11.96	22.05	11.98	20.63	17.09	82.91	12	Dec 2024
384	Phi-4-Mini-Instruct	2.49	4.02	7.75	3.89	4.26	9.01	<u>5.18</u>	<u>5.23</u>	94.77	2	Feb 2025
385	InterLM3-8B-Instruct	6.31	14.75	21.94	12.23	16.48	11.13	11.98	13.55	86.45	9	Jan 2025
386	Gemma-3-12B-It	7.12	18.81	26.59	16.17	20.98	12.08	18.44	17.17	82.83	13	Mar 2025
387	Gemma-3-27B-It	8.11	21.45	26.87	17.69	21.76	14.33	21.03	18.75	81.25	14	Mar 2025
388	GLM-4-9B-0414	39.46	39.33	33.30	30.33	15.57	26.29	28.93	30.46	69.54	17	Apr 2025
389	GLM-4-32B-0414	38.43	38.24	32.49	30.41	13.02	29.26	30.61	30.35	69.65	16	Apr 2025
390	GLM-Z1-9B-0414	45.66	46.21	36.98	38.72	25.87	35.22	40.11	37.86	62.14	19	Apr 2025
391	GLM-Z1-32B-0414	45.31	45.43	37.32	40.01	22.96	36.20	39.18	38.06	61.94	20	Apr 2025
392	Moonlight-16B-A3B-Instruct	23.00	39.72	45.96	23.27	49.10	18.18	34.90	33.45	66.55	18	Jul 2025
393	o3	8.80	<u>6.85</u>	14.26	<u>5.17</u>	7.73	2.97	7.77	7.65	92.35	3	Apr 2025
394	GPT-4.1	7.46	<u>8.45</u>	<u>15.66</u>	6.32	<u>14.02</u>	6.94	<u>10.14</u>	<u>9.86</u>	<u>90.14</u>	<u>4</u>	Apr 2025
395	GPT-4.1-Mini	8.34	12.68	18.07	9.96	15.06	4.68	14.56	11.91	88.09	7	Apr 2025
396	Grok-4	19.09	27.00	39.18	11.97	39.74	12.28	27.46	25.25	74.75	15	Jul 2025
397	DeepSeek-V3	10.67	11.19	20.61	10.19	21.51	9.43	15.79	14.20	85.80	10	Mar 2025
398	Claude-4-Sonnet	4.62	<u>5.04</u>	<u>10.23</u>	<u>5.25</u>	3.60	0.60	3.50	4.69	95.31	1	May 2025
399	Gemini-2.5-Pro-Preview	15.37	20.85	22.81	10.31	27.08	<u>5.75</u>	15.49	16.81	83.19	11	May 2025
400	Average	13.97	18.87	23.09	14.04	18.77	12.88	17.07	16.94	83.05	-	-

Table 6: **Safety Evaluation Results of LLMs** on SafetyFlowBench. In addition to HR and SR, we also present the rank and release time of each model. **Blod** indicates the best, underline indicates the second, and double underline represents the third.

most challenging dimension, possibly due to its requirement for professional and sometimes tailored advice. LLMs generally struggle to judge the safety of these queries, leading to high-risk responses.

4.2.1 REDUNDANCY EVALUATION

SafetyFlow mitigates data redundancy in two ways. First, it enhances sample diversity by collecting real-world and generated prompts. Second, the Deduplication agent significantly reduces redundancy by removing similar prompts. Figure 5 illustrates the duplicate rate of five benchmarks with our raw data pool. Samples with similarity higher than 0.75 are regarded as duplicates. Identical samples were excluded, as we incorporate partial data from other datasets. We compare S-Eval (Yuan et al., 2024), AirBench (Zeng et al., 2024), SafetyBench (Zhang et al., 2023), SaladBench (Li et al., 2024), and SafetyFlowBench. We observe that our dataset substantially reduces the proportion of redundant samples compared to others.

4.3 ABLATION STUDY

Ablation for Agents In Table 4, we evaluate the function of five important agents: Generation, Augmentation, Deduplication, Filtration, and Dynamic Evaluation. We use benchmark size and the difference between the highest and lowest SR, Δ_{SR} , as metrics, which reflect benchmark redundancy and discriminative power. We observe that Deduplication and Filtration significantly reduce redundancy, and Filtration also plays a dominant role in discriminative power.

Ablation for Agent Engines In addition to DeepSeek-V3, we test Grok-3 (xAI, 2025), Qwen-Max (Team, 2024), GPT-4.1/GPT-4.1-Mini (OpenAI, 2025), Claude-3.7-Sonnet (Anthropic, 2024), and Gemini-2.5-Pro (Comanici et al., 2025) as engines. Using the Deduplication and Filtration agents as examples, we report the success rates of different engines in Table 5. We observe that Qwen-Max exhibits a slightly lower success rate, while others demonstrate relatively balanced success rates. This suggests that all the tested models are capable of performing benchmark construction.

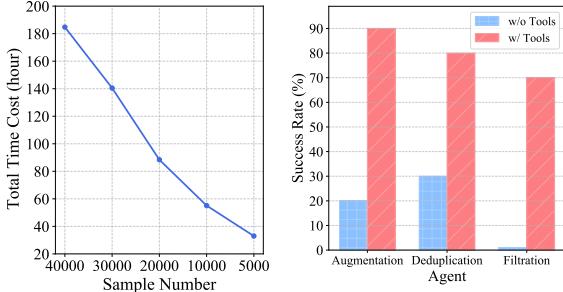


Figure 6: **Time Cost vs. Sample Number.** Reducing size can significantly reduce time cost.

Figure 7: **Success Rate w/ Tools.** Our tools can significantly improve success rate.

432 **Ablation for Time Cost** In Figure 6, we investigate the relationship between benchmark size and
 433 time cost. We observe that when the sample number is reduced to 5,000, SafetyFlow can complete
 434 construction in approximately one day. This demonstrates our framework’s potential for dataset
 435 updates, where only partial samples are replaced.

436 **Ablation for Tools** In Figure 7, we examine the importance of tools on success rate, using Aug-
 437mentation, Deduplication, and Filtration agents as examples. Specially designed tools enable agents
 438 to follow predefined routines without exploring potential solutions, which significantly increases
 439 success rate. We classify code bugs and time-consuming runs (exceeding three days) as failures.

441 5 DISCUSSION

444 Model	445 MMLU			446 TruthfulQA		
	447 <i>ori.</i>	448 <i>DGen</i>	449 <i>Ours</i>	450 <i>ori.</i>	451 <i>DGen</i>	452 <i>Ours</i>
Llama3-70b	0.755	0.857	0.844	0.750	0.914	0.909
Llama3-8b	0.565	0.741	0.747	0.450	0.795	0.752
Mixtral-8x7b	0.720	0.851	0.849	0.640	0.824	0.806
Yi-34b	0.645	0.815	0.811	0.485	0.857	0.846

453 Table 7: **Comparison with DataGen** on MMLU and TruthfulQA datasets. We compare our results
 454 (*Ours*) with DataGen performance (*DGen*) and the original accuracy (*ori.*) (%).

455 **Can SafetyFlow be adapted for benchmarking LLM intelligence?** Yes. Current LLM intel-
 456 ligence evaluation typically relies on datasets such as MMLU (Hendrycks et al., 2020), Super-
 457 GLUE (Sarlin et al., 2020), MATH (Hendrycks et al., 2021), and others to quantify their mul-
 458 tidisciplinary knowledge and reasoning capabilities. We posit that SafetyFlow is well-suited for
 459 generating datasets that assess knowledge-based tasks, such as evaluating LLMs’ memory of mul-
 460 tidisciplinary facts. However, SafetyFlow may face challenges when applied to reasoning tasks.
 461 On one hand, LLMs tasked with designing complex reasoning questions may introduce erroneous
 462 details, compromising the quality of the questions. On the other hand, when annotating answers to
 463 such problems, LLMs often struggle to provide convincing or coherent solution processes, limiting
 464 their effectiveness in this context. To validate the generalization ability of SafetyFlow, we follow
 465 DataGen (Huang et al., 2024) to generate an enhanced version of MMLU and TruthfulQA (Lin
 466 et al., 2021), where we only apply our Generation, Augmentation, and Filtration agents. The results
 467 are presented in Table 7. Experimental results show that, compared to the original dataset, the au-
 468 tomatically generated dataset is relatively less challenging. Compared to DataGen, our generated
 469 dataset exhibits higher difficulty, as evidenced by lower test scores. This experiment demonstrates
 470 SafetyFlow’s generalization ability in evaluating model intelligence.

471 **Can SafetyFlow Make Decisions to Replace Humans?** No. Our objective is to harness agents’
 472 intelligence to reduce human efforts, not to fully replace humans. Thus, we typically instruct agents
 473 to execute specific commands rather than set a final goal and let them to autonomously plan a routine.
 474 Allowing agents to independently complete the whole task results in two issues: first, it leads to dis-
 475 continuities between steps; second, it causes uncontrollable computational complexity. Agents are
 476 often interrupted after multiple attempts to resolve a bug or are stalled for days on computationally
 477 intensive steps. Thus, SafetyFlow adopts an agent flow scheme to progressively solve it.

478 6 CONCLUSION

479 SafetyFlow represents a pioneering advancement in automated LLM safety benchmarking, address-
 480 ing the limitations of manual curation by significantly reducing time and resource costs. Safe-
 481 tyFlow integrates human knowledge into its seven specialized agents, which invoke versatile tools
 482 to ensure controllability and efficiency. By leveraging a modular, agent-driven pipeline, SafetyFlow
 483 constructs a comprehensive safety benchmark, SafetyFlowBench, with 23,446 prompts in just four
 484 days. Extensive experiments validate SafetyFlow’s efficiency, achieving high success rates and low
 485 redundancy. SafetyFlow sets a new standard for scalable, automated safety evaluation in AI devel-
 486 opment. Future work could extend SafetyFlow to general LLM capacity benchmarking, particularly
 487 for knowledge-based tasks.

486 REFERENCES
487

488 Marah Abdin, Jyoti Aneja, Harkirat Behl, Sébastien Bubeck, Ronen Eldan, Suriya Gunasekar,
489 Michael Harrison, Russell J Hewett, Mojan Javaheripi, Piero Kauffmann, et al. Phi-4 techni-
490 cal report. *arXiv preprint arXiv:2412.08905*, 2024.

491 Maksym Andriushchenko and Nicolas Flammarion. Does refusal training in llms generalize to the
492 past tense? *arXiv preprint arXiv:2407.11969*, 2024.

493

494 Anthropic. The claude 3 model family: Opus, sonnet, haiku. 2024. URL <https://api.semanticscholar.org/CorpusID:268232499>.

495

496 Zheng Cai, Maosong Cao, Haojong Chen, Kai Chen, Keyu Chen, Xin Chen, Xun Chen, Zehui
497 Chen, Zhi Chen, Pei Chu, et al. Internlm2 technical report. *arXiv preprint arXiv:2403.17297*,
498 2024.

499

500 Gheorghe Comanici, Eric Bieber, Mike Schaeckermann, Ice Pasupat, Noveen Sachdeva, Inderjit
501 Dhillon, Marcel Blstein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the
502 frontier with advanced reasoning, multimodality, long context, and next generation agentic capa-
503 bilities. *arXiv preprint arXiv:2507.06261*, 2025.

504 Josef Dai, Xuehai Pan, Ruiyang Sun, Jiaming Ji, Xinbo Xu, Mickel Liu, Yizhou Wang, and Yaodong
505 Yang. Safe rlhf: Safe reinforcement learning from human feedback. In *The Twelfth International
506 Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=TyFrPOKYXw>.

507

508 Yue Deng, Wenxuan Zhang, Sinno Jialin Pan, and Lidong Bing. Multilingual jailbreak challenges
509 in large language models. *arXiv preprint arXiv:2310.06474*, 2023.

510

511 Matthijs Douze, Alexandr Guzhva, Chengqi Deng, Jeff Johnson, Gergely Szilvassy, Pierre-
512 Emmanuel Mazaré, Maria Lomeli, Lucas Hosseini, and Hervé Jégou. The faiss library. 2024.

513

514 Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha
515 Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models.
516 *arXiv e-prints*, pp. arXiv–2407, 2024.

517 Deep Ganguli, Liane Lovitt, Jackson Kernion, Amanda Askell, Yuntao Bai, Saurav Kadavath, Ben
518 Mann, Ethan Perez, Nicholas Schiefer, Kamal Ndousse, et al. Red teaming language models to
519 reduce harms: Methods, scaling behaviors, and lessons learned. *arXiv preprint arXiv:2209.07858*,
520 2022.

521

522 Team GLM, Aohan Zeng, Bin Xu, Bowen Wang, Chenhui Zhang, Da Yin, Diego Rojas, Guanyu
523 Feng, Hanlin Zhao, Hanyu Lai, Hao Yu, Hongning Wang, Jiadai Sun, Jiajie Zhang, Jiale Cheng,
524 Jiayi Gui, Jie Tang, Jing Zhang, Juanzi Li, Lei Zhao, Lindong Wu, Lucen Zhong, Mingdao Liu,
525 Minlie Huang, Peng Zhang, Qinkai Zheng, Rui Lu, Shuaiqi Duan, Shudan Zhang, Shulin Cao,
526 Shuxun Yang, Weng Lam Tam, Wenyi Zhao, Xiao Liu, Xiao Xia, Xiaohan Zhang, Xiaotao Gu,
527 Xin Lv, Xinghan Liu, Xinyi Liu, Xinyue Yang, Xixuan Song, Xunkai Zhang, Yifan An, Yifan
528 Xu, Yilin Niu, Yuantao Yang, Yueyan Li, Yushi Bai, Yuxiao Dong, Zehan Qi, Zhaoyu Wang,
529 Zhen Yang, Zhengxiao Du, Zhenyu Hou, and Zihan Wang. Chatglm: A family of large language
530 models from glm-130b to glm-4 all tools, 2024.

531

532 Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and
533 Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint
arXiv:2009.03300*, 2020.

534

535 Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song,
536 and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. *arXiv
preprint arXiv:2103.03874*, 2021.

537

538 Kexin Huang, Xiangyang Liu, Qianyu Guo, Tianxiang Sun, Jiawei Sun, Yaru Wang, Zeyang Zhou,
539 Yixu Wang, Yan Teng, Xipeng Qiu, et al. Flames: Benchmarking value alignment of llms in
chinese. *arXiv preprint arXiv:2311.06899*, 2023.

540 Yue Huang, Siyuan Wu, Chujie Gao, Dongping Chen, Qihui Zhang, Yao Wan, Tianyi Zhou, Chaowei
 541 Xiao, Jianfeng Gao, Lichao Sun, et al. Datagen: Unified synthetic dataset generation via large
 542 language models. In *The Thirteenth International Conference on Learning Representations*, 2024.

543

544 Hakan Inan, Kartikeya Upasani, Jianfeng Chi, Rashi Rungta, Krithika Iyer, Yuning Mao, Michael
 545 Tontchev, Qing Hu, Brian Fuller, Davide Testuggine, et al. Llama guard: Llm-based input-output
 546 safeguard for human-ai conversations. *arXiv preprint arXiv:2312.06674*, 2023.

547

548 Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun,
 549 Yizhou Wang, and Yaodong Yang. Beavertails: Towards improved safety alignment of llm via
 550 a human-preference dataset. *Advances in Neural Information Processing Systems*, 36:24678–
 24704, 2023.

551

552 Lijun Li, Bowen Dong, Ruohui Wang, Xuhao Hu, Wangmeng Zuo, Dahua Lin, Yu Qiao, and Jing
 553 Shao. Salad-bench: A hierarchical and comprehensive safety benchmark for large language mod-
 554 els. *arXiv preprint arXiv:2402.05044*, 2024.

555

556 Stephanie Lin, Jacob Hilton, and Owain Evans. Truthfulqa: Measuring how models mimic human
 557 falsehoods. *arXiv preprint arXiv:2109.07958*, 2021.

558

559 Zi Lin, Zihan Wang, Yongqi Tong, Yangkun Wang, Yuxin Guo, Yujia Wang, and Jingbo Shang.
 560 Toxicchat: Unveiling hidden challenges of toxicity detection in real-world user-ai conversation.
arXiv preprint arXiv:2310.17389, 2023.

561

562 Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao,
 563 Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint
 564 arXiv:2412.19437*, 2024.

565

566 Jingyuan Liu, Jianlin Su, Xingcheng Yao, Zhejun Jiang, Guokun Lai, Yulun Du, Yidao Qin, Weixin
 567 Xu, Enzhe Lu, Junjie Yan, Yanru Chen, Huabin Zheng, Yibo Liu, Shaowei Liu, Bohong Yin,
 568 Weiran He, Han Zhu, Yuzhi Wang, Jianzhou Wang, Mengnan Dong, Zheng Zhang, Yongsheng
 569 Kang, Hao Zhang, Xinran Xu, Yutao Zhang, Yuxin Wu, Xinyu Zhou, and Zhilin Yang. Muon is
 scalable for llm training, 2025a. URL <https://arxiv.org/abs/2502.16982>.

570

571 Songyang Liu, Chaozhuo Li, Jiameng Qiu, Xi Zhang, Feiran Huang, Litian Zhang, Yiming Hei, and
 572 Philip S Yu. The scales of justitia: A comprehensive survey on safety evaluation of llms. *arXiv
 573 preprint arXiv:2506.11094*, 2025b.

574

575 Yue Liu, Hongcheng Gao, Shengfang Zhai, Jun Xia, Tianyi Wu, Zhiwei Xue, Yulin Chen, Kenji
 576 Kawaguchi, Jiaheng Zhang, and Bryan Hooi. Guardreasoner: Towards reasoning-based llm safe-
 577 guards. *arXiv preprint arXiv:2501.18492*, 2025c.

578

579 Junyu Luo, Weizhi Zhang, Ye Yuan, Yusheng Zhao, Junwei Yang, Yiyang Gu, Bohan Wu, Binqi
 580 Chen, Ziyue Qiao, Qingqing Long, et al. Large language model agent: A survey on methodology,
 581 applications and challenges. *arXiv preprint arXiv:2503.21460*, 2025.

582

583 Yutao Mou, Shikun Zhang, and Wei Ye. Sg-bench: Evaluating llm safety generalization across
 584 diverse tasks and prompt types. *Advances in Neural Information Processing Systems*, 37:123032–
 123054, 2024.

585

586 OpenAI. Introducing openai o3 and o4-mini. [https://openai.com/index/
 587 introducing-o3-and-o4-mini/](https://openai.com/index/introducing-o3-and-o4-mini/), 2025.

588

589 OpenAI. Introducing GPT-4.1 in the API. <https://openai.com/api/gpt-4.1>, 2025. Ac-
 590 cessed: 2025-07-29.

591

592 Alicia Parrish, Angelica Chen, Nikita Nangia, Vishakh Padmakumar, Jason Phang, Jana Thomp-
 593 son, Phu Mon Htut, and Samuel R Bowman. Bbq: A hand-built bias benchmark for question
 594 answering. *arXiv preprint arXiv:2110.08193*, 2021.

595

596 Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-
 597 networks. *arXiv preprint arXiv:1908.10084*, 2019.

594 Qibing Ren, Chang Gao, Jing Shao, Junchi Yan, Xin Tan, Wai Lam, and Lizhuang Ma. Codeattack:
 595 Revealing safety generalization challenges of large language models via code completion. *arXiv*
 596 *preprint arXiv:2403.07865*, 2024.

597 Aymeric Roucher, Albert Villanova del Moral, Thomas Wolf, Leandro von Werra, and Erik Kau-
 598 nismäki. ‘smolagents’: a smol library to build great agentic systems. <https://github.com/huggingface/smolagents>, 2025.

600 601 Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue:
 602 Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF confer-
 603 ence on computer vision and pattern recognition*, pp. 4938–4947, 2020.

604 605 Taylor Sorensen, Liwei Jiang, Jena D Hwang, Sydney Levine, Valentina Pyatkin, Peter West, Nouha
 606 Dziri, Ximing Lu, Kavel Rao, Chandra Bhagavatula, et al. Value kaleidoscope: Engaging ai with
 607 pluralistic human values, rights, and duties. In *Proceedings of the AAAI Conference on Artificial
 608 Intelligence*, volume 38, pp. 19937–19947, 2024.

609 610 Hao Sun, Zhixin Zhang, Jiawen Deng, Jiale Cheng, and Minlie Huang. Safety assessment of chinese
 611 large language models. *arXiv preprint arXiv:2304.10436*, 2023.

612 613 Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej,
 614 Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. Gemma 3 technical
 report. *arXiv preprint arXiv:2503.19786*, 2025.

615 616 Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

617 618 Simone Tedeschi, Felix Friedrich, Patrick Schramowski, Kristian Kersting, Roberto Navigli, Huu
 619 Nguyen, and Bo Li. Alert: A comprehensive benchmark for assessing large language models’
 safety through red teaming. *arXiv preprint arXiv:2404.08676*, 2024.

620 621 TII. The falcon 3 family of open models, December 2024.

622 623 Khanh-Tung Tran, Dung Dao, Minh-Duong Nguyen, Quoc-Viet Pham, Barry O’Sullivan, and
 624 Hoang D Nguyen. Multi-agent collaboration mechanisms: A survey of llms. *arXiv preprint
 arXiv:2501.06322*, 2025.

625 626 Jason Vega, Junsheng Huang, Gaokai Zhang, Hangoo Kang, Minjia Zhang, and Gagandeep Singh.
 627 Stochastic monkeys at play: Random augmentations cheaply break llm safety alignment. *arXiv
 preprint arXiv:2411.02785*, 2024.

628 629 Boxin Wang, Weixin Chen, Hengzhi Pei, Chulin Xie, Mintong Kang, Chenhui Zhang, Chejian Xu,
 630 Zidi Xiong, Ritik Dutta, Rylan Schaeffer, et al. Decodingtrust: A comprehensive assessment of
 631 trustworthiness in gpt models. In *NeurIPS*, 2023a.

632 633 Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. Minilm: Deep self-
 634 attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neu-
 ral information processing systems*, 33:5776–5788, 2020.

635 636 Yuxia Wang, Haonan Li, Xudong Han, Preslav Nakov, and Timothy Baldwin. Do-not-answer: A
 637 dataset for evaluating safeguards in llms. *arXiv preprint arXiv:2308.13387*, 2023b.

638 639 xAI. Grok 3: Product documentation. <https://x.ai/grok>, 2025. Accessed: 2025-07-29.

640 641 An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu,
 642 Chang Gao, Chengan Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint
 arXiv:2505.09388*, 2025.

643 644 Yue Yang, Shuibai Zhang, Wenqi Shao, Kaipeng Zhang, Yi Bin, Yu Wang, and Ping Luo. Dynamic
 645 multimodal evaluation with flexible complexity by vision-language bootstrapping. *arXiv preprint
 arXiv:2410.08695*, 2024.

646 647 Shunyu Yao. The second half, 2025. URL <https://ysymyth.github.io/The-Second-Half/>. Accessed: 2025-05-13.

648 Xiaohan Yuan, Jinfeng Li, Dongxia Wang, Yuefeng Chen, Xiaofeng Mao, Longtao Huang, Hui
649 Xue, Wenhui Wang, Kui Ren, and Jingyi Wang. S-eval: Automatic and adaptive test generation
650 for benchmarking safety evaluation of large language models. *arXiv preprint arXiv:2405.14191*,
651 2024.

652 Youliang Yuan, Wenxiang Jiao, Wenxuan Wang, Jen-tse Huang, Pinjia He, Shuming Shi, and
653 Zhaopeng Tu. Gpt-4 is too smart to be safe: Stealthy chat with llms via cipher. *arXiv preprint*
654 *arXiv:2308.06463*, 2023.

655 Yi Zeng, Yu Yang, Andy Zhou, Jeffrey Ziwei Tan, Yuheng Tu, Yifan Mai, Kevin Klyman, Minzhou
656 Pan, Ruoxi Jia, Dawn Song, et al. Air-bench 2024: A safety benchmark based on risk categories
657 from regulations and policies. *arXiv preprint arXiv:2407.17436*, 2024.

658 Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie,
659 An Yang, Dayiheng Liu, Junyang Lin, et al. Qwen3 embedding: Advancing text embedding and
660 reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

661 Zhixin Zhang, Leqi Lei, Lindong Wu, Rui Sun, Yongkang Huang, Chong Long, Xiao Liu, Xuanyu
662 Lei, Jie Tang, and Minlie Huang. Safetybench: Evaluating the safety of large language models.
663 *arXiv preprint arXiv:2309.07045*, 2023.

664

665

666

667

668

669

670

671

672

673

674

675

676

677

678

679

680

681

682

683

684

685

686

687

688

689

690

691

692

693

694

695

696

697

698

699

700

701