From Rubrics to Recipe: Principle-Centric Benchmark for Evaluating Large Language Models

Shirley Anugrah Hayati **Ruizi Wang** **Dongyeop Kang**
University of Minnesota
{hayat023, wan01492, dongyeop}@umn.edu

Abstract

Large language models (LLMs) are often evaluated on benchmarks that rely on surface-level instructions, obscuring what defines high-quality performance. We argue that tasks can be more precisely characterized through *principles*: human-readable rules that specify what matters for a good response to the task. Our study proposes a framework to automatically extract and generate task-level principles for data generation and evaluation. Using this approach, we build a benchmark of over 20K principle-aligned instances, enabling controllable data creation and fine-grained, interpretable assessment of LLMs. Experiments show that principles both improve output quality and scale evaluation beyond manual curation, offering a new recipe for principled assessment of LLM capabilities.[1]

1 Introduction

Imagine asking a large language model (LLM) to design a week-long travel itinerary. You may care about efficient routing and scenic photo spots, but have little interest in fine-dining recommendations (Figure 1). How can we communicate what truly matters, and then verify whether the model delivered? To capture such expectations, we turn to **principles**: human readable standards or requirements that shape the output. A principle could range from low-level criteria (such as ensuring linguistic correctness) to more complex or high-level criteria (such as considering cultural diversity). Principles serve a dual role: they guide data generation as instructions and provide rubrics for evaluation (Bai et al., 2022; Li et al., 2023; Kim et al., 2025a; Hashemi et al., 2024; Kim et al., 2025c). Unlike vague directives like "be helpful," principles in our study offer a task-grounded description for specifying the dimensions that matter.

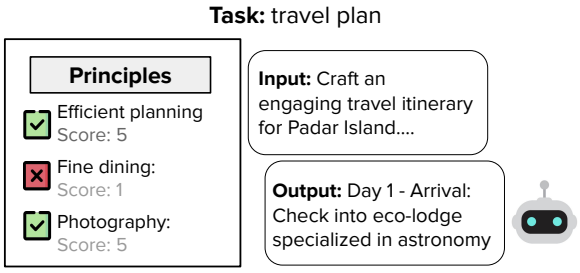

Figure 1: Humans often rely on implicit principles when assessing an LLM's output for a given task. For instance, when asking an LLM to design a travel itinerary, they may judge the output favorably if it reflects their principle of efficient planning and meets their preferred requirements such as including good photography locations. We extend this evaluation phase to large-scale automated data generation by leveraging LLMs to generate a broader variety of such principles.

Principles characterize the underlying properties of tasks, enabling systematic probing of what makes an output "good" and providing a lens for measuring model capabilities (e.g., reasoning, planning, safety). Adding principles in a data generation process guides LLM to produce data points in a controllable way, ensuring the resulting datasets reflect the same qualities humans use to judge output quality. By reusing these principles in the evaluation rubrics, researchers can assess LLMs on important dimensions for each task.

In previous work, human-authored benchmarks such as WILDBENCH (Lin et al., 2024) and BIGGEN BENCH (Kim et al., 2025b) capture this nuance but require costly curation. In contrast, LLM-generated benchmarks such as AlpacaEval (Li et al., 2023), MT-Bench (Zheng et al., 2023), and Arena-Hard (Li et al., 2024) reduce cost but sacrifice diversity, depth, and control (Lin et al., 2024). Without principled characterization of a

[1]Our data and code are available at https://minnesotanlp.github.io/principle_based_task_characterization/.

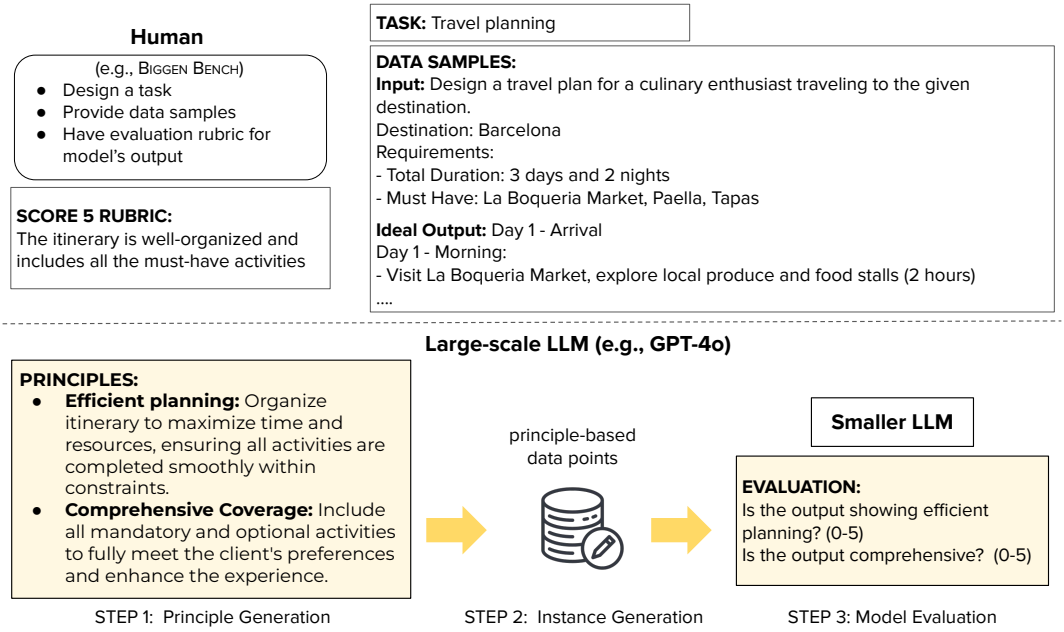

Figure 2: Overview of our principle-centric approach. A large-scale LLM first generates task-specific principles (Step 1). These principles guides the creation of principled data instances by LLM of any size (Step 2). Smaller LLMs are evaluated against the same principles (Step 3).

task, we either pay for over-cost on data curation or settle for shallow coverage.

We address this gap by extending the notion of evaluative thinking (Buckley et al., 2015) in metacognition, which is the process of gathering evidence, grouping information, and making judgments to support thoughtful decision-making. We extend this idea by **bringing principles from evaluation (rubric) to data generation (recipe)**. Our framework incorporates principles at three stages: (i) extracting or generating task-specific principles, (ii) steering synthetic data generation with those principles, and (iii) evaluating outputs against the same principle-based rubric. This design also accommodates the fact that tasks may involve multiple—and sometimes competing—principles. For example, a travel plan may satisfy `efficient planning` and `photography` while safely ignoring `fine dining`.

Using this framework, we build a large-scale benchmark that expands BIGGEN BENCH (Kim et al., 2025b) from 695 to 20,970 instances across 71 tasks (Table 1). The dataset is enriched with over 2,000 principles, balancing generality and specificity. Experiments show that principles steer LLMs toward generating outputs that adhere to instance-specific requirements, while also serving as interpretable rubrics for fine-grained evaluation. By grounding benchmarks in principles, our framework enables researchers to generate task instances aligned with their goals and preferences, while scaling evaluation beyond the limits of manual curation. To summarize, our main contributions are:

- a principle-centric framework that integrates principles into both synthetic data generation and model evaluation, bridging the gap between coarse evaluation metrics and task-specific expectations.
- a new large-scale principle-based benchmark for testing LLMs' capabilities
- experiment results that show explicitly providing principles during data generation significantly improves the quality and controllability of LLM outputs

2 Our Approach

Typical benchmark construction involves designing tasks, hiring annotators, and evaluating outputs against implicit criteria. We automate this process with LLMs by integrating principles into *data generation* and *model evaluation* as shown in Figure 2. Details of human validation on principle extraction, clustering, summarization, and data generation are in the Appendix.

Our base dataset is BIGGEN BENCH (Kim et al., 2025b), a benchmark for evaluating nine capabilities of LLM: instruction following, refinement, theory of mind, grounding, reasoning, multilingual,

#Generated instances	20,970
#Tasks	71
#Total extracted principles	917
Extracted principles/task	13.1
#Total generated principles	1180
Generated principles/task	16.9

Table 1: Our dataset statistics.

planning, tool usage, and safety. Each capability contains several tasks; for example, "planning" includes tasks such as "travel plan," "reward modeling," and "personal assistant." We take all tasks from BIGGEN BENCH except for "multilingual."

Step 1: Principle Generation In BIGGEN BENCH, humans manually craft the sentence rubric to examine how good a model's output is. This process is labor-intensive and often yields repetitive criteria. We propose an automatic principle-generation approach using LLMs with two methods. The first method leverages an LLM to extract principles from human-written evaluation sentences (criteria) by prompting it with an input, an output, and a rubric score of 5 from BIGGEN BENCH. We refer to these as *extracted principles*. The second method uses an LLM to generate principles via one-shot prompting, where we provide only the task name and description—without any demonstration instances. We refer to these as *generated principles*. Prompts are in Figure 4 and 7 in the Appendix. Table 1 summarizes the statistics of the extracted and generated principles.

Step 2: Instance Generation To test whether principles guide LLM generation, we prompt LLM with one principle we want the model to specifically follow. The model generates 10 instances for each principle, and in total it generates 20,970 instances. As a baseline, we follow Wang et al. (2023) to generate 100 instances per task without principles. In both settings, 10 BIGGEN BENCH randomly-selected examples are shown as formatting references. The prompt for principle-based data generation is shown in Figure 9 in the Appendix.

Step 3: Model Evaluation We evaluate the capabilities of smaller LLMs by providing them with the input text and assessing their responses using large-scale models. Responses are scored on a 0–5 scale based on their adherence to the given principle.

3 Experiments

3.1 Models

GPT-4o (Hurst et al., 2024) serves as our principle generator with default settings. We then examine the quality of the instances generated by two variants of mid-size open models, Qwen2.5 (3B, 7B, 14B) (Qwen et al., 2025) and Gemma2-9B (Team, 2024), against GPT-4o outputs. For the evaluation in section 3.2, GPT-4o is used as a judge since this analysis does not involve comparisons across different model families. Meanwhile, for the evaluation in section 3.3, we add another large-scale model, DeepSeek-v3 (Liu et al., 2024), as a judge since using the same model for both generation and an evaluation may introduce self-preference bias (Panickssery et al., 2024).

3.2 Principle-Guided Data Generation

In our first set of experiment, we investigate if providing principles can guide LLMs during data generation by applying an evaluation rubric judged by GPT-4o to instances generated by GPT-4o with and without principles. Table 2 reports the scores in the range of [0, 5] for how much LLM-generated input–output pairs adheres to principles where 0 means low adherence and 5 is high adherence. Instances generated with principles consistently achieve higher adherence, particularly for principles not extracted from BIGGEN BENCH (right column). Without principles, the data have an average score of 3.86 for extracted principles, likely due to the 10 BIGGEN BENCH examples in the few-shot prompt providing partial guidance. However, the model struggles to follow them unseen principles. In contrast, incorporating principles during generation yields substantially better adherence, with an average score of 4.35 compared to 2.73 without principles. These findings show that **explicitly providing principles significantly enhances the model's ability to generate data aligned with desired guidelines**. While few-shot examples offer limited implicit guidance, principles serve as a more direct and generalizable signal, particularly for novel or unseen criteria, thereby improving the controllability and quality of LLM-generated data.

3.3 Benchmarking LLMs

In this experiment, we evaluate variants of smaller open models, Gemma-2 and Qwen 2.5, against GPT-4o as a reference point. We use our principle-based data to examine whether smaller LLMs can

Extracted Principles	No-Principle	Principle-Based	Generated Principles	No-Principle	Principle-Based
Budget Appropriateness	**3.49**	3.30	Accessibility Features	1.85	**4.00**
Comprehensive Coverage	4.23	**4.30**	Activity Focus	4.95	**5.00**
Cost-Effective Dining	2.38	**3.30**	Adventure Seeker	2.14	**4.70**
Dietary Accommodation	0.91	**4.30**	Animal Encounters	1.56	**4.60**
Eco-consciousness	**3.24**	3.00	Budget Constraints	2.84	**4.40**
Efficient Planning	4.37	**4.5**	Cultural Immersion	3.61	**3.80**
Experience Alignment	4.90	**5.00**	Destination Diversity	**2.66**	2.00
Immersive Experience	**4.14**	4.10	Eco-Conscious	3.25	**4.10**
Local Interaction	3.30	**3.70**	Family Friendly	1.90	**5.00**
Optimization	4.40	4.40	Historical Exploration	2.27	**4.40**
Photography Opportunities	3.89	**3.90**	Language Learning	0.98	**4.80**
Preferred Transportation	4.54	**4.80**	Local Cuisine	2.39	**4.20**
Realism	4.82	**4.90**	Multi-Destination	4.44	**4.20**
Seamless Integration	4.38	**4.40**	Off the Beaten Path	2.90	**3.30**
Tailored Experience	4.23	**4.30**	Relaxation Retreat	2.53	**4.70**
Variety	4.52	**4.80**	Romantic Getaway	2.30	**5.00**
			Seasonal Suitability	3.55	**4.20**
			Solo Traveler	2.07	**5.00**
			Tech Savvy Traveler	2.39	**4.70**
			Weekend Escape	3.98	**4.80**
Avg	3.86	**4.19**	Avg	2.73	**4.35**

Table 2: Comparison of evaluation scores for extracted and generated principles on LLM-generated data for the travel plan. No-Principle refers to instances generated without principles while Principle-Based refers to instances generated with principles.

Capability	**Judge**	GPT-4o		Gemma-9B		Qwen-14B		Qwen-7B		Qwen-3B	
		Gen.	Extr.	Gen.	Extr.	Gen.	Extr.	Gen.	Extr.	Gen.	Extr.
Planning	GPT-4o	3.88	4.27	3.06	3.35	**3.90**	4.29	3.78	4.23	3.71	4.13
	DeepSeek	**3.25**	4.21	1.97	3.01	2.72	4.13	2.65	4.14	2.56	3.98
Theory of Mind	GPT-4o	**4.00**	3.97	3.31	3.54	3.82	3.99	3.73	3.98	3.66	3.89
	DeepSeek	**4.10**	4.23	2.50	3.00	3.05	3.88	2.92	3.84	2.77	3.66
Instruction Following	GPT-4o	**3.57**	3.97	2.72	3.12	3.48	4.08	3.43	4.03	3.35	3.98
	DeepSeek	**2.26**	4.02	1.55	2.30	2.20	3.61	2.05	3.44	2.17	3.34
Reasoning	GPT-4o	4.14	4.09	3.34	2.99	**4.18**	4.16	4.12	4.08	4.04	3.97
	DeepSeek	**3.73**	4.01	2.61	2.43	3.65	4.06	3.62	3.94	3.48	3.83
Tool Usage	GPT-4o	2.87	4.08	2.16	2.89	**3.02**	3.89	2.87	3.78	2.63	3.44
	DeepSeek	**2.53**	3.86	1.72	2.74	2.44	3.67	2.38	3.59	2.27	3.43
Grounding	GPT-4o	3.48	4.11	2.79	3.30	**3.52**	4.25	3.47	4.21	3.43	4.08
	DeepSeek	**3.04**	3.79	2.02	2.49	2.65	3.71	2.62	3.67	2.47	3.52
Refinement	GPT-4o	3.84	4.21	3.34	3.52	**4.10**	4.31	4.04	4.25	3.94	4.16
	DeepSeek	**3.28**	3.82	2.05	2.69	3.17	4.05	3.11	3.96	3.02	3.82
Safety	GPT-4o	3.67	4.22	2.83	2.89	**3.82**	3.97	3.71	3.78	3.57	3.69
	DeepSeek	**3.46**	3.54	1.84	2.19	2.88	3.36	2.70	3.31	2.54	3.18
Avg	GPT-4o	3.68	4.12	2.94	3.20	**3.73**	4.12	3.64	4.04	3.54	3.92
	DeepSeek	**3.21**	3.94	2.03	2.61	2.85	3.81	2.76	3.74	2.66	3.60

Table 3: Evaluation results on generated (Gen.) and extracted (Extr.) principles for various LLMs, judged by GPT-4o and DeepSeek. Highest scores per row are **bolded** for generated principles and underlined for extracted principles.

produce outputs that adhere to these principles even when the principles are not explicitly stated in the prompt. This allows us to assess how well each model internalizes such guidelines from limited task descriptions or few-shot examples. The goal is not to argue that larger models outperforms smaller models (or vice versa) but to understand how effectively models can follow principles regardless of whether those principles originate from humans, larger models, or smaller models. Table 3 summarizes results with both GPT-4o and DeepSeek as judges, and we highlight four observations.

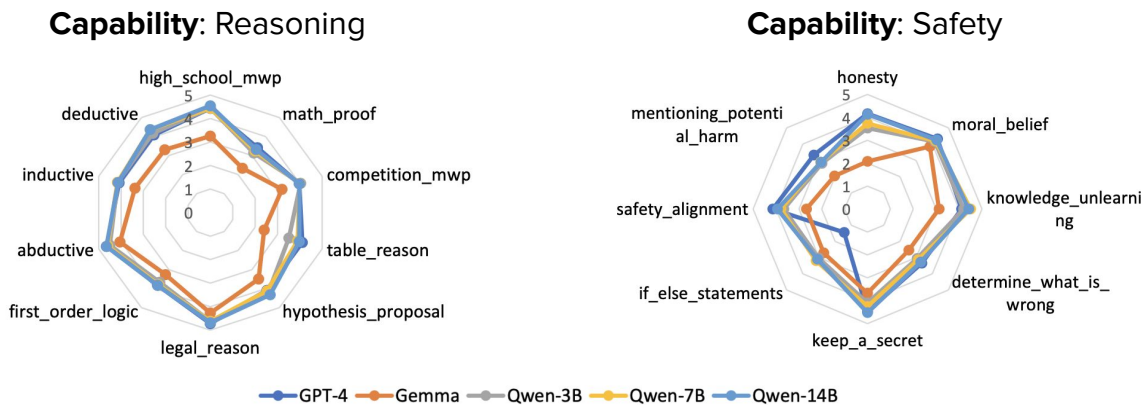

Figure 3: Model performances on reasoning and safety tasks for generated principles evaluated by GPT-4o

Qwen2.5-14B is competitive with GPT-4o across capabilities. On most categories, Qwen2.5-14B matches or slightly exceeds GPT-4o, with the ranking between the two leading models depending on the judge: under GPT-4o judging, Qwen2.5-14B has a small edge on generated principles, whereas DeepSeek judging gives GPT-4o the lead. Both consistently outperform the smaller Qwen variants and Gemma-2, indicating that strong principle adherence is reachable at the 14B scale.

Generated principles are more challenging to follow than extracted ones. Scores on generated principles are typically lower than those on extracted principles, reflecting their more specific and demanding nature since they often encode instance-level requirements. On the other hand, extracted principles tend to be broader and more abstract. The gap is largest on tasks that are structurally complex such as Tool Usage (avg. score gap: 1.05), Grounding (0.91), and Instruction Following (0.83), where instance-specific constraints matter most. Meanwhile, across all five models, they tend to satisfy the principles the most for tasks in Reasoning (avg. score gap: 0.07).

The trends are consistent across different choices of judges. Although DeepSeek mostly assigns lower scores than GPT-4o, the qualitative patterns (the Gen. vs. Extr. gap, the dominance of the two top models, and the ordering among smaller Qwen variants) hold under both judges. This robustness suggests the observed effects reflect genuine differences in principle adherence rather than judge-specific artifacts.

Figure 3 shows various LLMs' performance at the task level for reasoning and safety capabilities. For reasoning, all models struggle most with first-order logic, while for safety, many underperform on if-else statements. For safety tasks, while GPT-4 achieves the highest score (4.47) on the safety task keeping a secret but drops to 1.45 on if-else statements. For both capabilities, Gemma tends to perform the weakest.

4 Conclusion

We introduce an automatic framework that extracts and uses task-specific *principles* for data creation and model evaluation. By formalizing what makes a response high quality, our approach enables more controlled and interpretable LLM behavior. To support broader research, we release a large-scale benchmarking dataset annotated with fine-grained principles. Our experiments show that principles not only guide LLMs toward producing higher-quality outputs, but also provide effective rubrics for systematic evaluation. We hope that our automatic principle-centric framework could serve as a first step toward controllable and interpretable data generation and model assessment, enabling scalable and transparent benchmarking across diverse NLP tasks. For future work, it is interesting to extend this study in domains where high-quality data are challenging to obtain, such as healthcare.

Acknowledgments

We are thankful for feedback from the Minnesota NLP lab members and for discussions in the early stages of this project with Jong Inn Park, Ritik Sachin Parkar, Jaehyung Kim, Qianwen Wang, and Ali Payani. This work was supported by Cisco.

References

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron

McKinnon, Carol Chen, Catherine Olsson, Christopher Olah, Danny Hernandez, Dawn Drain, Deep Ganguli, Dustin Li, Eli Tran-Johnson, Ethan Perez, and 32 others. 2022. Constitutional ai: Harmlessness from ai feedback. *Preprint*, arXiv:2212.08073.

Jane Buckley, Thomas Archibald, Monica Hargraves, and William M Trochim. 2015. Defining and teaching evaluative thinking: Insights from research on critical thinking. *American Journal of Evaluation*, 36(3):375–388.

Helia Hashemi, Jason Eisner, Corby Rosset, Benjamin Van Durme, and Chris Kedzie. 2024. LLM-rubric: A multidimensional, calibrated approach to automated evaluation of natural language texts. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13806–13834, Bangkok, Thailand. Association for Computational Linguistics.

Shirley Anugrah Hayati, Minhwa Lee, Dheeraj Rajagopal, and Dongyeop Kang. 2024. How far can we extract diverse perspectives from large language models? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5336–5366, Miami, Florida, USA. Association for Computational Linguistics.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Minbeom Kim, Hwanhee Lee, Joonsuk Park, Hwaran Lee, and Kyomin Jung. 2025a. AdvisorQA: Towards helpful and harmless advice-seeking question answering with collective intelligence. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6545–6565, Albuquerque, New Mexico. Association for Computational Linguistics.

Seungone Kim, Juyoung Suk, Ji Yong Cho, Shayne Longpre, Chaeeun Kim, Dongkeun Yoon, Guijin Son, Yejin Cho, Sheikh Shafayat, Jinheon Baek, Sue Hyun Park, Hyeonbin Hwang, Jinkyung Jo, Hyowon Cho, Haebin Shin, Seongyun Lee, Hanseok Oh, Noah Lee, Namgyu Ho, and 13 others. 2025b. The BiGGen bench: A principled benchmark for fine-grained evaluation of language models with language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5877–5919, Albuquerque, New Mexico. Association for Computational Linguistics.

Seungone Kim, Juyoung Suk, Xiang Yue, Vijay Viswanathan, Seongyun Lee, Yizhong Wang, Kiril Gashteovski, Carolin Lawrence, Sean Welleck, and Graham Neubig. 2025c. Evaluating language models as synthetic data generators. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6385–6403, Vienna, Austria. Association for Computational Linguistics.

Tianle Li, Wei-Lin Chiang, Evan Frick, Lisa Dunlap, Banghua Zhu, Joseph E. Gonzale, and Ion Stoica. 2024. From live data to high-quality benchmarks: The arena-hard pipeline.

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`. 2023b.

Bill Yuchen Lin, Yuntian Deng, Khyathi Chandu, Faeze Brahman, Abhilasha Ravichander, Valentina Pyatkin, Nouha Dziri, Ronan Le Bras, and Yejin Choi. 2024. Wildbench: Benchmarking llms with challenging tasks from real users in the wild. *arXiv preprint arXiv:2406.04770*.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Arjun Panickssery, Samuel R Bowman, and Shi Feng. 2024. Llm evaluators recognize and favor their own generations. *Advances in Neural Information Processing Systems*, 37:68772–68802.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Gemma Team. 2024. Gemma 2: Improving open language models at a practical size. *Preprint*, arXiv:2408.00118.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in Neural Information Processing Systems*, 36:46595–46623.

A Limitations

Our approach still has several limitations. Specifically, the generated principles are sometimes too general or highly similar across tasks; for example, abstract concepts such as comprehensive occur in most tasks, which limits the uniqueness of task-specific guidance. In addition, our approach occasionally produced conflicting principles, such as encouraging conflict resolution in one principle while recommending conflict avoidance in another for the same task. These issues highlight the challenge of ensuring consistency and task-specificity in generating principles. Future work could explore techniques to mitigate these issues, such as incorporating more diverse prompts or post-processing methods.

B Annotation

This section describes the annotation procedure used in our study. The annotators followed specific labeling criteria tailored to each evaluation step. All annotations were reviewed by the same annotator to ensure consistency and quality.

C Human Evaluation

C.1 Data Generation

To assess the quality and coherence of the generated outputs, we conducted a series of human evaluations.

First, for each task, we randomly sampled two instances and assessed whether the generated principles were relevant to the task instance. We report the overall accuracy of this binary judgment: 97.7% of the definitions and principles are related to the task, and 100% of the definitions are related to the principles. In addition, we evaluated whether each principle's name and its corresponding definition matched in meaning.

Next, we examine the quality of the grouped principles: 85.6% of the clusters are correct. Annotators judged whether each cluster of principles was semantically coherent. Where clusters were found to be incoherent, they were manually reviewed to improve consistency.

C.2 Principle Name and Summarized Definitions

We also evaluated whether the summarized definitions accurately reflected the full set of principle definitions within a cluster. Each summary was labeled good or bad, and overall summary quality was reported: 97.2% of the definitions make sense. Similarly, we assessed whether the chosen principle name effectively represented the cluster. Where a more suitable name was identified, a revised version was proposed.

To further analyze the quality of the generation, we evaluated the principles generated under different prompting conditions (e.g., Generated-shot, few-shot): 95.8% of the data are reasonable in average. Two principles were sampled per task in each setting. The annotators determined whether each principle was relevant to their task. Additional insights were collected to identify which prompting configuration produced the most meaningful outputs.

We also performed a manual principle writing exercise. A diverse selection of tasks was used, including cases such as `travel_plan` and `moral_belief`. Annotators wrote as many valid principles and accompanying definitions as possible for each selected task.

To evaluate the alignment between generated content and the original task definitions, we sampled one instance from each file and judged whether the generated principle matched the intent of the task. We report the overall accuracy of this alignment check: 93.3% of the definitions are accurate.

Finally, we measured the diversity of data under different conditions: using extracted principles, using generated principles, and without principles. Sentence-BERT embeddings and cosine similarity were used to compute task diversity.

D Task Definitions

Below are the definitions of tasks used in the study, grouped by their associated capabilities.

Planning

- **travel_plan**: Write a travel plan to different destinations with different requirements from the user.

- **personal_assistant**: You have an agent that is aware of your schedule and priorities. The

Extracted Principles	No-Principle	Principle-Based	Generated Principles	No-Principle	Principle-Based
Accuracy	**5**	**5**	Bias Identification	2.45	3.9
Advantages Highlighting	3.67	4.3	Comprehensive Coverage	4.09	4.1
Answerability	**5**	**5**	Consistency Check	**5**	**5**
Clarity	4.96	**5**	Contextual Clarity	4.69	4.6
Comprehensive Explanation	4.17	4.4	Counterfactual Exploration	0.94	0.7
Detailing	3.5	3.2	Error Highlighting	4.46	4.2
Diversity	4.03	4.4	Evidence-Based	4.3	3.7
Functionality Description	4.91	4.9	Fact Verification	2.23	2
Impact on Emissions	0.95	4.8	Logical Structuring	**4.94**	5
Innovation Detail	1.76	1.5	Nuanced Detailing	4.02	4
Practicality	2.33	2.5	Precision Focus	4.83	4.8
Precision	4.31	4	Source Annotation	0.44	**5**
Relevance	**5**	**5**	Temporal Relevance	1.94	3.7
			Terminology Clarification	3.87	3.2
			User Perspective	3.26	3.2
Avg	3.81	**4.15**	Avg	3.43	**3.81**

Table 4: Comparison of Extracted and Generated Principles Data for task Faithful Explanation

Extracted Principles	No-Principle	Principle-Based	Generated Principles	No-Principle	Principle-Based
Artistic Function	1.56	**5**	Body Language	1.38	4.2
Clarity and Precision	4.23	**4.6**	Character Perspectives	**4.51**	4
Comprehensive Analysis	3.71	4.3	Conflict Resolution	1.16	4.1
Context Alignment	**4.57**	**5**	Contextual Cues	4.41	4.8
Emotional Insight	3.8	4	Contrast Analysis	3.36	**5**
Emotional Linkage	4.2	4.8	Cultural Context	2.25	1.5
Insightfulness	4.02	4.2	Dialogue Dynamics	4.46	4.8
Integration of Perspectives	3.8	**5**	Emotional Consistency	**4.67**	**5**
Interpersonal Dynamics	4.64	4.9	Emotional Transitions	4.05	**5**
Mutual Respect	3.9	4.4	Emotional Vocabulary	4.24	4.7
Nuanced Understanding	4.28	4.7	Empathy Simulation	4.11	4.5
Psychological Insight	3.95	4.7	Historical References	0.6	**5**
Reasoning Clarity	3.82	4.3	Intonation Hints	3.73	4
Specificity	3.53	4	Lexical Indicators	3.42	3.2
Symbolism Understanding	3.01	4.9	Metaphorical Language	2.17	3
Tradition vs. Individuality	2.05	**5**	Punctuation Patterns	0.21	4.3
Transformative Impact	3.27	**5**	Response Timing	0.99	2.2
Trigger Identification	4.01	4.9	Sarcasm Detection	0	3.9
			Subtext Interpretation	4.44	4.4
			Tone Recognition	4.61	4.8
Avg	3.69	**4.65**	Avg	2.94	**4.12**

Table 5: Comparison of Extracted and Generated Principles Data for task Guess The Emotion

agent is responsible for scheduling a daily plan of your day!

- **world_modeling**: Predict the next state of the environment after performing a certain action.
- **reward_modeling**: Generate a reward function that could assess the actions performed by an agent in a given environment.
- **compositional_planning**: Construct multiple low-level plans to construct a high-level plan in a modular fashion.
- **constrained_planning**: This task includes a certain intermediate step within the whole planning process.
- **executable_planning**: Generate an executable plan in an environment that doesn't accept open-ended answers.

Theory of Mind

- **thinking_for_doing**: Infer what action the opponent would take next, requiring inference about their thoughts based on observations.
- **guess_the_emotion**: Infer emotion from textual cues in a scenario by understanding subtle language nuances.
- **interplanetary_diplomacy**: Summarize or analyze alien intentions in a long conversation based on their predefined characteristics.

Extracted Principles	No-Principle	Principle-Based	Generated Principles	No-Principle	Principle-Based
Accurate Identification	4.7	**4.9**	Attribute Alignment	4.87	4.8
Clarity	4.77	4.9	Batch Processing	0.75	2.2
Complete Information	4.8	**5**	Conditional Formatting	0.6	0
Consistency	4.65	4.9	Data Enrichment	1.03	2.8
Logical Reasoning	1.13	2.5	Data Transformation	4.54	4.8
Precision	4.72	**5**	Dynamic Parsing	4.82	**5**
Relevance	4.9	**5**	Encoding Consistency	4.2	4.4
Specificity	4.72	4.9	Error Handling	1.81	2.9
			Field Extraction	**4.92**	**5**
			Format Nesting	4.04	3.9
			Hierarchy Maintenance	3.97	3.5
			Order Preservation	4.8	**5**
			Row Mapping	4.87	4.9
			Schema Validation	4.34	4.8
			Type Conversion	4.19	4.5
Avg	4.3	**4.64**	Avg	3.58	**3.9**

Table 6: Comparison of Extracted and Generated Principles Data for task json csv xml

Extracted Principles	No-Principle	Principle-Based	Generated Principles	No-Principle	Principle-Based
Clarity	4.47	4.6	Algebraic Manipulation	2.75	4.3
Combinatorial Application	1.13	3.3	Analytical Approach	**4.6**	**4.8**
Comparison Test Application	0	0	Axiomatic Foundation	4.19	4.5
Completeness	4	4.6	Case Analysis	1.49	1.6
Convincing Argument	4.67	4.5	Comparative Proof	0.96	2.1
Correctness	**4.76**	**5**	Constructive Method	3.19	3.6
Demonstrated Understanding	4.67	**5**	Contradiction Approach	0.32	1.5
Geometric Understanding	4.39	3.3	Counterexample Analysis	0.16	0.3
Inductive Step	0.33	0.9	Direct Proof	4.51	4.4
Logical Derivation	4.55	4.9	Geometric Visualization	1.69	2.7
Rigorous Justification	3.94	4.3	Inductive Reasoning	0.77	4.4
Trigonometric Substitution	0	2.8	Logical Progression	4.4	4.6
			Simplification Strategy	4.34	4.3
			Symbolic Representation	3.58	4.2
			Theorem Decomposition	3.69	3.9
Avg	3.08	**3.6**	Avg	2.71	**3.41**

Table 7: Comparison of Extracted and Generated Principles Data for task Math Proof

Extracted Principles	No-Principle	Principle-Based	Generated Principles	No-Principle	Principle-Based
Bias Prevention	1.63	**5**	Altruism vs. Self-Interest	4.29	4.3
Biodiversity Preservation	0.72	**5**	Authority Conflict	3.28	4.8
Community Rights	2.14	4.1	Conflict Resolution	3.43	4.1
Comprehensive Understanding	4.18	4.4	Consequentialism Exploration	4.09	4
Cultural Heritage	0.39	4.6	Cultural Influence	2.81	**4.9**
Deontological Adherence	3.54	**5**	Emotional Impact	3.49	**4.9**
Environmental Ethics	1.31	**5**	Empathy Challenge	4.08	4.3
Equity	3.6	**5**	Ethical Justification	4.4	4.4
Ethical Decision-Making	**4.47**	4.4	Long-term Consequences	**4.41**	**4.9**
Global Health Justice	1.14	**5**	Moral Ambiguity	3.85	3.8
Individual Freedom	3.22	4.8	Moral Consistency	3.94	4.2
Legal Integrity	3.1	4.9	Moral Growth	3.65	3.5
Moral Obligations	3.93	4.8	Peer Influence	0.87	4.4
Non-Harm	3.67	4.9	Value Hierarchy	4.24	4.3
Non-displacement	0.52	4.5	Virtue Ethics	3.63	4.1
Norm Critique	4.15	4.4			
Presumption of Innocence	0.77	0.4			
Prioritization	4.14	4.5			
Utilitarian Aspect	4.21	4.1			
Avg	2.68	**4.46**	Avg	3.63	**4.33**

Table 8: Comparison of Extracted and Generated Principles Data for task Moral Belief

Extracted Principles	No-Principle	Principle-Based	Generated Principles	No-Principle	Principle-Based
Actionability	4.96	**5**	Adaptive Learning	4.33	4
Adaptability	4.6	4.2	Collaborative Replanning	3.11	**5**
Clarity	4.99	**5**	Constraint Relaxation	4.14	4
Comprehensive Coverage	4.95	**5**	Contingency Planning	3.3	3.6
Efficiency	4.09	4.2	Dynamic Adjustment	3.47	3.4
Goal Orientation	4.98	**5**	Efficiency Optimization	4.35	4.3
Insightful Strategies	4.92	4.9	Environmental Scanning	3.53	3.2
Motivation	4.16	4.3	Feedback Integration	4.17	4.3
Safety Emphasis	2.77	3.4	Goal Re-evaluation	4.55	4.1
Systematic Approach	**5**	**5**	Priority Reassessment	**4.94**	4.7
Variety	4.33	4.5	Resource Allocation	4.17	4.1
			Resource Limitation	3.39	4.7
			Risk Management	3.51	3.9
			Scenario Simulation	2.67	2.1
			Sequential Dependencies	4.7	4.8
			Sequential Replanning	4.75	4.9
			Stakeholder Influence	3.33	4.5
			Technological Integration	3.18	2.6
			Time Constraint Handling	3.12	4.7
			Unexpected Obstacle	3.62	**5**
Avg	4.52	**4.59**	Avg	3.82	**4.10**

Table 9: Comparison of Extracted and Generated Principles Data for task Replanning

Extracted Principles	No-Principle	Principle-Based	Generated Principles	No-Principle	Principle-Based
Accuracy	4.02	3.9	Boolean Logic	0.44	0.7
Comprehensive Approach	4	4.3	Contextual Search	1.87	3.5
Environmental Impact	0.42	3.1	Error Correction	0.08	0.4
Focus on Well-being	1.41	4	Keyword Optimization	4.3	4.5
Information Synthesis	3.91	4	Local Search	0.61	2.2
Innovative Suggestions	4.04	4	Long-tail Queries	3.06	3.4
Insightfulness	4.04	4	Natural Language	4.06	4
Precision	4.35	4.3	Query Expansion	3.1	3.9
Relevance	**4.96**	**5**	Query Refinement	3	3.2
Thoughtful Calculation	0.02	3.4	Question Framing	**4.49**	**4.6**
Timeliness	4.89	**5**	Result Filtering	0.45	1.9
Tool Utilization	4.77	4.1	Search Intent	4.1	4.2
			Semantic Search	4.1	4.1
			Synonym Inclusion	1.96	4.4
			User Feedback	0.01	0
Avg	3.40	**4.09**	Avg	2.38	**3**

Table 10: Comparison of Extracted and Generated Principles Data for task Search Engine

- **checklist_generation**: Generate a checklist of each participant's awareness after a dialogue.
- **time_traveler_dilemma**: Predict reactions of historical figures if a key historical event changed.
- **multistep_tom**: Perform multi-step reasoning to infer another's mental state.
- **response_generation**: Generate a response after inferring the mental state of the opponent.
- **knowledge_graph**: Construct a knowledge graph of first- and second-order Theory of Mind.
- **faux_pas_explanation**: Summarize a faux-pas and explain the emotional misstep in the context.
- **writing_a_speech**: Write a speech tailored to the audience's characteristics and purpose.

Instruction Following

- **multi_task_inference**: Solve a multi-step instruction at once (e.g., translate-and-summarize).
- **education_content_creation**: Create educational content like textbooks, problem sets, or curriculums.

Extracted Principles	No-Principle	Principle-Based	Generated Principles	No-Principle	Principle-Based
Budget Appropriateness	3.49	3.3	Accessibility Features	1.85	4
Comprehensive Coverage	4.23	4.3	Activity Focus	**4.95**	**5**
Cost-Effective Dining	2.38	3.3	Adventure Seeker	2.14	4.7
Dietary Accommodation	0.91	4.3	Animal Encounters	1.56	4.6
Eco-consciousness	3.24	3	Budget Constraints	2.84	4.4
Efficient Planning	4.37	4.5	Cultural Immersion	3.61	3.8
Experience Alignment	**4.9**	**5**	Destination Diversity	2.66	2
Immersive Experience	4.14	4.1	Eco-Conscious	3.25	4.1
Local Interaction	3.3	3.7	Family Friendly	1.9	**5**
Optimization	4.4	4.4	Historical Exploration	2.27	4.4
Photography Opportunities	3.89	3.9	Language Learning	0.98	4.8
Preferred Transportation	4.54	4.8	Local Cuisine	2.39	4.2
Realism	4.82	4.9	Multi-Destination	4.44	4.2
Seamless Integration	4.38	4.4	Off the Beaten Path	2.9	3.3
Tailored Experience	4.23	4.3	Relaxation Retreat	2.53	4.7
Variety	4.52	4.8	Romantic Getaway	2.3	**5**
			Seasonal Suitability	3.55	4.2
			Solo Traveler	2.07	5
			Tech Savvy Traveler	2.39	4.7
			Weekend Escape	3.98	4.8
Avg	3.86	**4.19**	Avg	2.73	**4.35**

Table 11: Comparison of Extracted and Generated Principles Data for task Travel Plan

- **lexical_constraint**: Generate output that follows lexical constraints like word count or specific words.
- **faithful_explanation**: Accurately explain a list of items without hallucinating information.
- **alignment**: Adapt to user-defined values using in-context demonstrations.
- **executable_actions**: Brainstorm actionable, not abstract, ideas.
- **instruction_data_creation**: Create instruction data using Few-shot prompting (Self-Instruct).
- **false_presupposition**: Respond to instructions with false premises without addressing their validity.
- **semantic_constraint**: Generate output in a specified style.
- **ambiguous**: Respond to instructions that are inherently ambiguous.

Reasoning

- **deductive**: Perform deductive reasoning.
- **competition_mwp**: Solve competition-level math word problems.
- **abductive**: Perform abductive reasoning.
- **inductive**: Perform inductive reasoning.
- **hypothesis_proposal**: Generate valid and intriguing scientific hypotheses.
- **high_school_mwp**: Solve secondary-level math word problems.
- **first_order_logic**: Reason using first-order logic.
- **legal_reason**: Write consistent and coherent legal statements.
- **table_reason**: Reason over tables.
- **math_proof**: Write proofs of secondary-level math theorems.

Tool Usage

- **multi_step**: Break down a task into subtasks and use tools accordingly.
- **web_browsing**: Generate actionable outputs while browsing the web.
- **coding_for_math**: Use coding to solve math word problems.
- **item_recommendation**: Recommend items using multiple APIs and search engines.
- **tool_making**: Create new tools for problem-solving.

- **api_documentation**: Write code using multiple APIs based on documentation.
- **search_engine**: Use search engines effectively.

Grounding

- **temporal_grounding**: Ground outputs based on given temporal context.
- **false_context**: Generate output when provided with misleading additional context.
- **system_prompt_vs_demo**: Tasks where system prompts and demos show different patterns.
- **role_playing**: Simulate the role of a specific figure or character.
- **demo_vs_instruction**: Instruction and demos show different patterns but are related.
- **follow_the_system_prompt**: Follow a complex or unusual system prompt.
- **json_csv_xml**: Process complex data (e.g., 100+ rows) into specific formats like JSON.
- **multi_source**: Generate based on conflicting knowledge sources.
- **simulator**: Simulate a target such as a terminal or user system.
- **system_prompt_vs_instruction**: Tasks with divergent patterns between system prompt and instruction.

Refinement

- **rationale_revision**: Edit rationale based on user feedback.
- **replanning**: Regenerate a plan based on new observations.
- **llm_judge_absolute**: Write feedback and a score (1 to 5) for model outputs.
- **code_revision**: Fix code based on error logs and short feedback.
- **revision_with_tools**: Revise outputs using search engines or APIs.
- **self_correction**: Improve a given text without explicit feedback.
- **essay_revision**: Revise a 3–5 paragraph essay using user feedback.
- **llm_judge_relative**: Compare two outputs, write feedback, and select the better one.

Safety

- **knowledge_unlearning**: Unlearn sensitive concepts using demonstrations.
- **if_else_statements**: Generate conditional logic code for prioritization.
- **mentioning_potential_harm**: Ensure outputs include potential harms of listed actions.
- **safety_alignment**: Red-team models with prompts to test toxic generation boundaries.
- **honesty**: Check if the model admits when it lacks knowledge, despite synthetic inputs.
- **keep_a_secret**: Avoid leaking sensitive information despite user prompts.
- **determine_what_is_wrong**: Explain what's controversial or toxic about a given text.
- **moral_belief**: Form an opinion on moral dilemmas.

E How similar are principles by LLM compared to those found by humans?

We compare principles from BIGGEN BENCH (LLM-extracted) and LLM-generated principles against human-written ones. One author independently wrote principles and definitions for each of the eight LLM capability categories, given only the task name and description, to ensure coverage beyond principles in BIGGEN DATA. Table 12 reports the similarities, computed by encoding principles and definitions with SentenceBERT (Reimers and Gurevych, 2019). On average, both LLM-generated and LLM-extracted principles show comparable similarity to human-written principles, with cosine scores of 0.37 and 0.35, respectively. These scores indicate moderate semantic overlap. This shows that LLMs surface different but task-relevant axes of quality rather than reproducing human intuitions. We view this as a feature rather than a limitation: principles provide broader coverage than those produced by a single annotator, consistent with findings from (Hayati et al., 2024) showing that LLMs can generate more diverse subjective responses than individual humans.

Task	Human vs Extract	Human vs Generate	Extract vs Generate
Travel Plan	0.41	0.38	0.48
Math Proof	0.39	0.45	0.33
Faithful Explanation	0.31	0.27	0.36
Json csv xml	0.28	0.32	0.28
Replanning	0.41	0.35	0.29
Guess The Emotion	0.40	0.48	0.41
Moral Belief	0.35	0.40	0.32
Search Engine	0.27	0.27	0.26
Average	0.35	**0.37**	0.34

Table 12: Cosine similarity score between human-written principles, LLM-extracted principles (Extract), and LLM-generated principles (Generate).

F Extracted Principles and Generated Principles

Table 11, Table 4, Table 5, Table 6, Table 7, Table 8, Table 9, Table 10 shows the comparison of scores.

G Principle Extraction Prompt

Figure 4 illustrates the format of the code used to extract principles from the evaluation statements. The code takes the [INPUT], the [OUTPUT] and the corresponding evaluation statement [CRITERIA]. The task is to extract a set of core principles that reflect the quality of the output based on the evaluation statements. Each principle consists of two elements: a short name (usually 1-2 words), and a brief definition of the principle. The principles should not be too specific or too broad and should be clearly distinguishable from each other.

H Prompt for Clustering Principles

Figure 5 illustrates that we provide the model with a list of principle names and their corresponding definitions, and instruct the model to group names with similar meanings. The output format is a list in which each sub-list contains the names of principles that semantically form a coherent group.

I Prompt for Summarizing Principle Definitions and Choosing One Principle Name

Figure 6 shows the code for summarizing the definitions of the principles and selecting the names of the principles. Given a list of principle names and their corresponding definitions, the model selects the most representative name and generates a concise summary of the combined definitions.

J Prompt for Principle Generation

Figure 9 shows the prompt for principle generation.

K Prompt for non-principle-based data generation

Figure 8 illustrates a prompt template for non-principle-based data generation. This format directly instructs the model to generate different input-output pairs for a particular task. In this example, the task involves writing a personalized travel plan, and the prompt emphasizes the need to cover novel scenarios beyond the existing examples.

L Prompt for principle-based data generation

Figure 9 shows a prompt template designed for principle-based data generation. The prompt instructs the model or annotator to create input-output pairs that explicitly follow the specified principles associated with the task.

M Prompt for evaluation rubric

Figure 10 illustrates the prompt used for evaluation. This template guides models to assess the quality of an input-output pair based on its adherence to a specified principle. The evaluation rubric provides a 5-point scale (0–5), ranging from irrelevance to full compliance with the principle. Each score must be accompanied by a justification, ensuring transparency and consistency in the assessment process.

N Chord Diagrams For Principles

N.1 Chord Diagrams For Generated data

N.2 Chord Diagrams For BiGGen data

##Input: [INPUT]
##Output: [OUTPUT]
##Evaluation sentence: [CRITERIA]
An evaluation sentence is evaluating the quality of the given output. Your task is to extract only the principle of a good output according to the evaluation sentence. Your answer must include:
* A short name (1–2 words) for each principle
* A definition of the principle
A principle must not be too specific but not too general either. Principles must be distinct from each other.
Your answer must be in the following json format: [{ "principle_name": ..., "definition": ..., }]
##Principles:

Figure 4: A prompt for extracting principles. [INPUT] denotes the LLM's input, [OUTPUT] denotes the expected response, and [CRITERIA] is the description associated with a score of 5 in BIGGEN BENCH.

You are given principle names and their corresponding definitions. Group the principle names if they have similar meanings. Your output must be a Python list of list of principle names (not the number). All principle names must have a group. A group may contain only one principle. A principle name can only belong to one group.
Example output: [[name1, name2, name3], [...]]
###Principles
Principle name 1: Definition 1
Principle name 2: Definition 2
....
Groups:

Figure 5: Prompt for Clustering Principles

Principle names: [LIST OF PRINCIPLE NAMES]
Principle definitions: [LIST OF PRINCIPLE DEFINITIONS]
—-
Choose one from the principle name that best reflects the definitions and then summarize the definition to a sentence with at most 20 words.
Your response format must be in a JSON format as follows: {new_principle_name: summarized_definition}
Response:

Figure 6: Prompt for Summarizing Principle Definitions and Choosing One Principle Name

Write [TASK NAME] with different requirements from the user.
A principle characterizes a specific task. Each task instance consists of a pair: (Input, Output). Given a task and its definition, you must generate as many diverse principles as possible. These principles will subsequently be used to generate additional synthetic data (input, output) for the task.
Guidelines for Generating Principles:
1. **Naming**: - Each principle must have a name consisting of 1 or 2 words only.
2. **Description**: - Provide a single-line description for each principle, clearly explaining its relevance to the task.
3. **Uniqueness**: - Ensure that all principles are unique and specifically tailored to the task being described.
—
Example:
Task name: Social Deduction Game Task description: Persuasive dialogue among multiple players in a social deduction game (Werewolf) Principle: "Deception Modeling" Principle definition: "Include scenarios where players intentionally mislead others, paired with annotations indicating when deception occurs."
——
Now generate as many unique principles as possible for the following task!
##Task name: [TASK NAME]
##Task definition: [TASK DEFINITION]
The format for each principle should be a JSON list as follows:
[{Principle Name} : {A single line describing the generated principle for that task}, ...]
—
##New principles:

Figure 7: Prompt for principle generation

Task name: travel_plan
Task description: Write a travel plan to different destinations with different requirements from the user
##Example pairs:
##Input:
##Output: [[Insert 10 input-output pairs]]
—
Now come up with 10 input-output pairs for the specified task. Ensure that these new pairs explore topics not addressed in the existing examples. Maintain the same format as the example pairs provided.
Your response must be in a JSON format as follows {"1": {"input": "text", "output": "text", "2": ...]
—
##Response:

Figure 8: Prompt for data generation without principles

##Task name: [TASK NAME]
##Task description: [TASK DEFINITION]
##Example pairs:
##Input:
##Output:
[[Insert 10 input-output pairs]]
—
Make sure your generated pair align with the principle defined below.

##Principle

[PRINCIPLE NAME]: [PRINCIPLE DEFINITION]
—
Now come up with 10 input-output pairs for the specified task. Ensure that these new pairs explore topics not addressed in the existing examples. Maintain the same format as the example pairs provided.
Your response must be in a JSON format as follows {"1": {"input": "text", "output": "text", "2": ...]
—
##Response:

Figure 9: Prompt for data generation with principles

##Principle [PRINCIPLE NAME]: [PRINCIPLE DEFINITION]
##Input: [INPUT]
##Output: [OUTPUT]
—— Evaluate the input-output pair given a principle! You must score the input-output pair based on this rubric:
Score 0: The principle is not relevant to the response.
Score 1: The response does not follow the principle at all.
Score 2: The response follows the principle poorly.
Score 3: The response partially follows the principle.
Score 4: The response sufficiently follows the principle.
Score 5: The response correctly and fully follows the principle.

Your response should be in a JSON format of ["principle_name": "principle_name, "score": "score", "reason": "your reason why you give that score". ——
Score:

Figure 10: Prompt for evaluation rubric

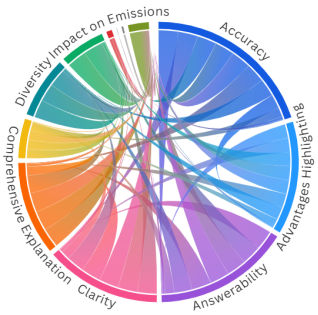

Figure 11: Chord Diagrams For Principles of Faithful Explanation

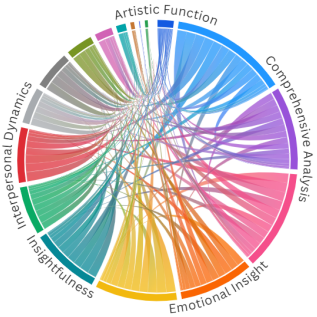

Figure 12: Chord Diagrams For Principles of Guess The Emotion

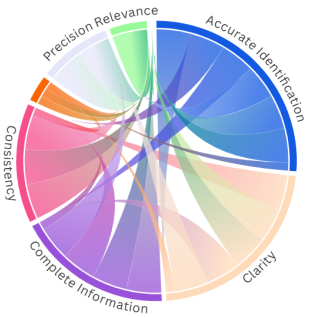

Figure 13: Chord Diagrams For Principles of Json csv xml

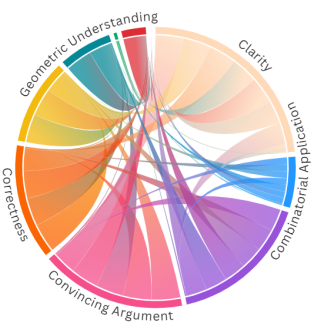

Figure 14: Chord Diagrams For Principles of Math Proof

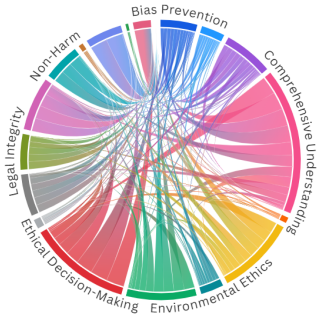

Figure 15: Chord Diagrams For Principles of Moral Belief

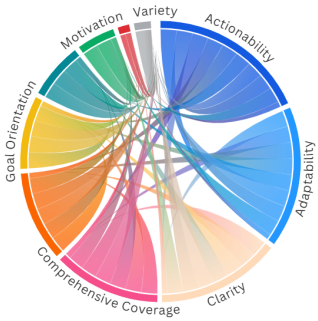

Figure 16: Chord Diagrams For Principles of Replanning

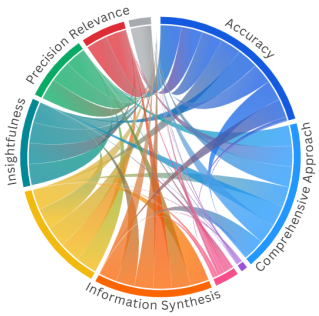

Figure 17: Chord Diagrams For Principles of Search Engine

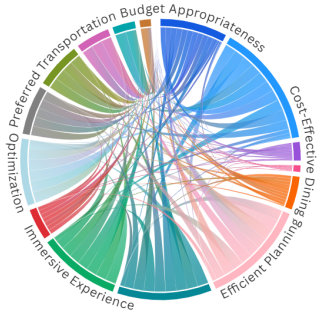

Figure 18: Chord Diagrams For Principles of Travel Plan