CDT: A Comprehensive Capability Framework for Large Language Models Across Cognition, Domain, and Task

Anonymous ACL submission

Abstract

Recent advances in Large Language Models (LLMs) have significantly enhanced their capabilities, highlighting the need for comprehensive evaluation frameworks that extend beyond task-specific benchmarks. However, existing benchmarks often focus on isolated abilities, lacking a holistic framework for assessing LLM capabilities. To address this gap, we propose the **C**ognition-**D**omain-**T**ask (CDT) framework, which comprehensively measures a model's capabilities across three dimensions. We expand the scope of model capability definitions at the cognitive level by incorporating the Cattell-Horn-Carroll cognitive theory, refining the categorization of model capabilities. In addition, we propose two data selection methods based on this framework, which has shown significant improvements in both general and specific benchmarks. These results demonstrate the effectiveness of our CDT framework and its practical utility. Source code and model will be available at `https://anonymous.4open.science/r/CDT-641F`.

1 Introduction

Recent advances in Large Language Models (LLMs) have significantly expanded their capabilities. The introduction of reinforcement learning (Kumar et al., 2024; Wang et al., 2024a; Hu et al., 2023) and chain-of-thought reasoning (Wei et al., 2022; Wang et al., 2023a) has further enhanced their reasoning abilities. Notable LLMs such as OpenAI's o1 (OpenAI, 2024b) and DeepSeek R1 (DeepSeek-AI, 2025) have demonstrated remarkable reasoning capabilities. As LLMs become more sophisticated, accurately evaluating their underlying abilities is increasingly crucial. Current benchmarks, such as MMLU (Hendrycks et al., 2021), AlpacaEval (Dubois et al., 2024), and GSM8K (Cobbe et al., 2021), are widely used to assess these capabilities.

Framework	Open Source Tagging Models	Multiple Dimensions	Capability Decomposition	Cognition Oriented	Domain Oriented	Task Oriented
FLASK	✗	✓	✗	✓	✓	✗
FAC^{2}E	✗	✓	✓	✓	✗	✓
INSTAG	✓	✗	✗	✗	✓	✗
CDT (Ours)	✓	✓	✓	✓	✓	✓

Table 1: Comparison between our LLM capability frameworks with INSTAG (Lu et al., 2024), FLASK (Ye et al., 2024b), and FAC^{2}E (Wang et al., 2024b). Our CDT framework addresses the gaps and limitations of existing methods across multiple dimensions.

However, many of them focus on isolated aspects of model capabilities, such as coding, commonsense reasoning, or specific task performance, and the ability dimensions are always task-oriented and limited, without a holistic framework that systematically categorizes and defines the full spectrum of LLM capabilities. For instance, benchmarks like MMLU evaluate knowledge mastery across academic disciplines but overlook dimensions like code generation. Prior work, including Zhong et al. (2025), highlights that evaluations tend to neglect the interplay of multiple abilities. Recent efforts like FLASK (Ye et al., 2024b) and FAC^{2}E (Flanagan et al., 2000) focus on multi-model comparisons but fall short in capability decomposition and multi-dimensional analysis. Additionally, while works like INSTAG (Lu et al., 2024) explore capability applications, definitions remain underdeveloped. Those works raise the fundamental question: *What core capabilities constitute an effective large language model?* We propose the **C**ognition-**D**omain-**T**ask (CDT), a meticulously structured multi-dimensional model capability framework, to address this question.

Our proposed CDT capability framework is a comprehensive taxonomy for categorizing and decomposing LLM capabilities across three dimensions: cognition, domain, and task. At the cognitive level, based on the Cattell-Horn-Carroll (CHC) theory (Schneider and McGrew, 2018), a foundational framework in cognitive science, we select 16 core cognitive abilities that are most relevant and suitable for LLMs, providing precise definitions for each. At the domain level, we identify nine domain scenarios commonly encountered by LLMs and further refine these into 33 distinct subdomains. At the task level, drawing inspiration from prior work on dataset construction (Wang et al., 2022, 2023b; Ouyang et al., 2022), we systematically categorize task types across diverse instructions, culminating in a taxonomy of 13 task types. We conduct a comparative analysis between existing capability frameworks and our proposed CDT framework, with the results summarized in Table 1.

After constructing the CDT framework, we extend its application to LLMs, beginning with data selection. We propose simple yet effective selection methods tailored to both diversity-driven general scenario and capability-oriented specific scenario. For the general scenario, we introduce a diversity-driven data selection approach, ensuring that the training data encompasses a broad spectrum of capabilities. For the specific scenario, we propose a capability-oriented data selection method by leveraging small amounts of specific data to identify and extract the requisite capabilities, which subsequently inform the data selection process. In the diversity-driven general scenario and capability-oriented specific scenario, our data selection methods achieve average scores of 42.2 and 66.7, respectively. These results significantly outperform other capability-related methods and baseline approaches. Our **main contributions** are as follows:

- We propose CDT, a comprehensive framework that systematically categorizes LLMs' abilities across cognition, domain, and task.
- We develop specialized tag models for each dimension to enable fine-grained tagging of capacities at the instruction level.
- We investigate the application of the CDT framework in data selection for both diversity-driven general scenario and capability-oriented specific scenario, proposing methods that lead to significant improvements in model capabilities.
- We will release all the data, tag models, and training scripts used in our CDT framework.

2 Related Works

Definitions of LLMs' Capability Research on defining LLM capabilities primarily falls into two categories. The first approach focuses on integrating capabilities with data, where models adjust data distributions to optimize learning (Nottingham et al., 2024; Polo et al., 2025; Chen et al., 2023; Wu et al., 2024; Lu et al., 2024). For instance, Chen et al. (2023) propose a method for data allocation based on an ordered skill set, where skills are represented as a directed graph, suggesting that mastering one skill facilitates the acquisition of others. However, this method is dataset-specific and lacks a universal definition of model capabilities. Similarly, Wu et al. (2024) introduce an MLP-based scoring network to guide data allocation, framing fine-tuning as a bi-level optimization problem. This approach treats different datasets as representing distinct capabilities. The second approach defines model capabilities from task-specific and domain-specific perspectives, often relying on labeled data for evaluation. Zhong et al. (2025) present a hierarchical framework of model capabilities, encompassing seven foundational and seven complex abilities derived from their interrelationships. Similarly, Ye et al. (2024b) analyze open-source LLMs to identify four key capabilities, further subdividing them into 12 fine-grained skills, thereby offering a comprehensive evaluation framework.

Applications of LLMs' Capability A primary application of capability frameworks is the development of evaluation benchmarks for large models. Additionally, research is exploring how these evaluations can inform the construction of more capable models. For domain-specific evaluations, Xia et al. (2024) propose FoFo, a framework that assesses LLMs' capabilities across multiple domains based on their format-following ability. For general capability evaluation, Hendrycks et al. (2021), Dubois et al. (2024), and Srivastava et al. (2022) have advanced benchmarks for assessing broad model competencies. Zhong et al. (2025) evaluate model capabilities by leveraging carefully designed prompts within their capability framework. Similarly, Ye et al. (2024b) use annotated test instructions to assess LLM performance on a defined capability scale, scoring models based on both responses and instruction alignment. In terms of enhancing domain-specific capabilities, Wang et al. (2024c) propose Re-Task, which integrates capability frameworks with Chain-of-Thought (COT) to decompose tasks and enhance subtask-specific abilities. Lee et al. (2024) introduce THANOS, a multi-turn dialogue dataset that improves model performance by breaking down conversational capabilities. Xu et al. (2023) present LaRS, which improves CoT reasoning by selecting data with similar capabilities, fostering better reasoning abilities.

3 Method

3.1 Capability Framework Construction

In the CDT framework we develop, we define model capabilities from three perspectives: cognition, domain, and task. While the domain and task perspectives have been extensively explored in recent research, we build upon this foundation with adjustments to better capture their nuances. From the cognition perspective, we define capabilities through the lens of the CHC theory in cognitive science. The CHC theory, grounded in earlier explorations of human cognition (Carroll, 2003; Cattell, 1963; Horn, 1965; Flanagan et al., 2000), serves as a foundational model in cognitive science (McGrew and Evans, 2004). In the realm of computer science, numerous studies have demonstrated the critical role of cognitive capabilities in LLMs and artificial intelligence (Zhao et al., 2022; Lieto et al., 2018; Song et al., 2024). Our overall capability framework is shown in Figure 1.

Cognition The CHC theory categorizes human cognitive abilities into three hierarchical levels. Stratum I consists of "narrow" abilities, which represent specialized skills developed through experience, learning, or the application of targeted methodologies (Carroll, 1993). Stratum II encompasses "broad" abilities, which are more abstract and general in nature. Stratum III represents the highest level, with a single general cognitive ability acting as an overarching factor. In our framework, we focus exclusively on the Stratum I abilities defined by Schneider and McGrew (2018), as they provide specific abilities and detailed definitions that are more directly applicable than those found in the other two levels. The process of constructing LLM cognitive capabilities follows these steps:

- **Cognition Selection**: As the CHC theory models human cognitive abilities across multiple modalities, including vision, hearing, and speaking, we first filter out non-linguistic abilities to align with our focus on language models, leaving multimodal extensions for future work. Next, we remove skills that are essential for humans but not as crucial for models, such as memory-related abilities. Additionally, we exclude abilities tied to domain knowledge, as our framework already treats domain expertise as a separate dimension. After this filtering process, the number of abilities is reduced from 82 to 14.

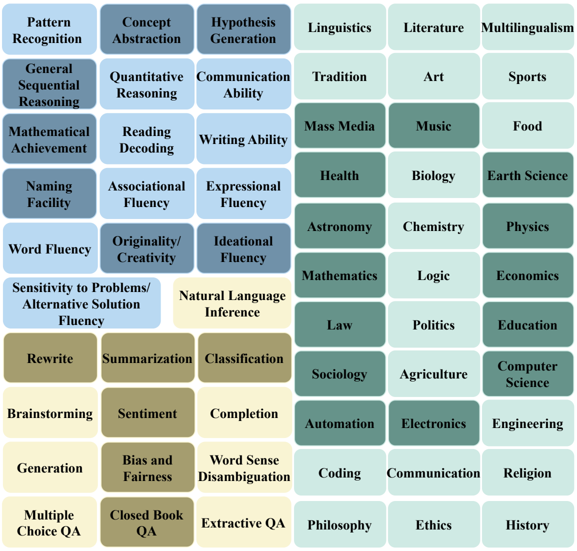


Figure 1: The model capability framework we define, where the blue section represents the Cognition dimension, the green section represents the Domain dimension, and the brown section represents the Task dimension.

- **Definition Refinement**: To better align with language models, we refine certain ability definitions. Notably, the ability Induction, originally defined as "the ability to discover the underlying characteristic (e.g., rule, concept, process, trend, class membership) that governs a problem or a set of materials," often leads to ambiguity in capability tagging. Its broad and abstract nature makes it frequently assigned across diverse instructions. To address this, we subdivide it into three specific capabilities: pattern recognition, concept abstraction, and hypothesis generation. After these refinements, the total number N_c of cognitive capabilities is 16. We define cognition dimension $\mathcal{C}$ as follows:

$$\mathcal{C} = \{c_i\}_{i=1}^{N_c} \tag{1}$$

where c_i is the specific cognitive capability.

Domain Based on Ye et al. (2024b), which categorizes 38 domains, we construct the domain dimension of our framework. However, we observe that certain domains, such as business and marketing, exhibit considerable similarity, potentially introducing ambiguity in capability tagging models and leading to label distribution dispersion. So, we manually refine the domain set, resulting in $N_d = 33$ domains in our framework. The domain dimension $\mathcal{D}$ can be formally expressed as:

$$\mathcal{D} = \{d_i\}_{i=1}^{N_d} \tag{2}$$

where d_i is the categorized domain.

Task For task categorization, inspired by Wang et al. (2022); Bach et al. (2022); Ouyang et al. (2022), we comprehensively consider task granularity and completeness, ultimately selecting $N_t = 13$ tasks. For task definition, we synthesize information from Wikipedia and prior work (Ding et al., 2023) to formulate detailed definitions for each task. The task dimension $\mathcal{T}$ is as follows:

$$\mathcal{T} = \{t_i\}_{i=1}^{N_t} \tag{3}$$

where t_i is the task we define.

Finally, the whole capability framework $\mathcal{F}$ is:

$$\mathcal{F} = \{(c, d, t) \mid c \in \mathcal{C},\ d \in \mathcal{D},\ t \in \mathcal{T}\} \tag{4}$$

Details on the categorization and definitions of each capability are provided in Appendix A.2.

3.2 Capability Tagging Model Training

To facilitate the practical use of our framework, we train a capability tagging model for each dimension. We prompt GPT-4o (OpenAI, 2024a) to annotate fine-grained capability tags for each query in the original seed data due to its exceptional ability to understand nuanced instructions and contextual relationships, making it highly effective for tagging tasks that require deep comprehension. Given the pivotal role of cognitive abilities in human intelligence, we assign up to two cognitive capabilities to each data point. In contrast, for the domain and task dimensions, only a single tag is assigned. Then we use the tagged training data $\mathcal{D}_t = \{(q_i, l_i)\}_{i=1}^{N}$, where N is the total amount of the training dataset, q is the query and l is the tagged capability labels, to train three capability annotators.

We use the dataset constructed by FLASK as our training set, which is derived from multiple high-quality NLP datasets. The dataset consists of 1,740 samples, and we randomly split 10% of the data into a test set to evaluate annotator performance. We design our prompts following the approaches proposed by Lu et al. (2024); Ye et al. (2024b). To mitigate position bias, we randomize the order of capabilities in the prompt for each data point. Additionally, when tagging cognitive capabilities, we ask the models to generate an explanation paired with each tag, as cognitive tasks require a deeper understanding of the instructions. All prompts are presented in Appendix A.1. We fine-tune the Qwen2.5-7B-Base (Team, 2024) model for 120 steps, evaluate it every 40 steps, and select the checkpoint with the best performance. The training is configured with a batch size of 32 and a cosine learning rate schedule set to 2e-5.

To validate the performance of the trained annotators, we use the GPT-generated labels as the ground truth and evaluate the models on the test set. The accuracy rates for cognition, domain, and task tags are 93%, 78%, and 77%, respectively, yielding an average score of 82.7%. For the cognition tagging task, since we assign two tags to each data point, a match is considered correct if at least one tag matches when calculating accuracy. In comparison, the annotator from INSTAG achieves a performance of 73.4% on its test set. In terms of overall labeling accuracy, CDT significantly outperforms INSTAG, demonstrating that our method is more accurate in data annotation and less susceptible to confusion.

4 CDT For Dual-Scenario Application

While the CDT framework offers a comprehensive definition of model capabilities, its application to LLMs remains an area requiring further exploration. Leveraging CDT's ability to classify data at the instruction level based on capabilities, we focus on its application to data selection for LLM instruction fine-tuning. This approach enables the systematic enhancement of training data quality and relevance, ultimately improving LLM performance on downstream tasks. Prior to implementing the data selection process, we first annotate the collected data set D_{pool} using the CDT framework to ensure precise capability-based categorization, resulting in the labeled dataset $D^{'}_{pool}$. We then define the capability composites within $D^{'}_{pool}$ as T_d.

$$T_d = \text{Composites}(D^{'}_{pool}) \tag{5}$$

where Composites means getting all the capability composites in a given labeled dataset.

4.1 Diversity-Driven General Scenario Data Selection

When training LLMs, data diversity plays a crucial role in enhancing model performance and generalization (Miranda et al., 2024; Zhou et al., 2023). Therefore, we propose a diversity-driven general data selection method based on CDT. Firstly, we define the selected training dataset as D_{train} and the composite capability assigned to D_{train} as T_s.

$$T_s = \text{Composites}(D_{train}) \tag{6}$$

For diversity-driven applications, our goal is to enlarge T_s as much as possible. Then we define a threshold R, which denotes the ratio of T_s to T_d, we quantify the attribute diversity:

$$R = \frac{|T_s|}{|T_d|} \tag{7}$$

where $|\cdot|$ denotes the cardinality (i.e., the number of elements) of a set. The value of R reflects the coverage rate of unique composite capabilities within the selected sub-dataset relative to the entire data pool. Our selection criterion aims to maximize the proximity of R to 1. Based on this, if a data point $d \in D_{pool}$ could increase R, we add the composite of d to T_s and d itself to D_{train} as training data. When R can no longer be increased, we perform an average selection from D_{pool} to fill the gaps in the capability composite of T_s.

4.2 Capability-Oriented Specific Scenario Data Selection

When applying the capability framework in the capability-oriented specific scenario, we first label the validation set of the test task to obtain the labeled dataset D_{valid}. Then, we tag D_{valid} with our annotators to form D'_{valid} and use the same method as in the diversity-driven approach to extract all combinations of abilities T_v from D'_{valid}.

$$T_v = \text{Composites}(D'_{valid}) \tag{8}$$

We aim to perform an average selection of the data from D'_{pool} based on the combinations of capabilities in T_v. However, in practice, T_v may be limited to a small subset of combinations of capabilities, and the amount of data corresponding to these combinations in D'_{pool} may not be sufficient to support our selection. To address this issue, we further decompose the capabilities in T_v. Specifically, we break down the triplet of capabilities $f = (c, d, t)$ into binary pairs $(c, d), (c, t), (d, t)$, creating a binary set T_v^*, and further into individual dimensions $(c), (d), (t)$, forming a unary set $T_v^\star$. When the triplet set T_v does not yield enough data, we first perform random selection on T_v^*, followed by selection on $T_v^\star$ in successive stages. This approach ensures sufficient data collection while preserving the concentration of capabilities. We present the details of the two algorithms in Appendix A.3.

5 Experiments

5.1 Experiment Setup

Data Pool and Base Model To evaluate and apply our proposed capability framework, CDT, across both diversity-driven general scenario and capability-oriented specific scenario, we utilize the following datasets: (1) Aggregated high-quality datasets, including Flan V2 (Longpre et al., 2023) and Chain of Thought (CoT) (Wei et al., 2022); and (2) Open-ended generation datasets with human-annotated responses, such as Dolly (Conover et al., 2023) and Open Assistant (Köpf et al., 2023). From these four datasets, we compile a pool of approximately 270,000 data points. These datasets are characterized by high complexity and generalization, which aligns well with the capability framework presented in this paper and establishes a solid foundation for subsequent experiments. Since our annotators are trained using Qwen2.5-7B[1], we select Llama2-7B-Base[2] as the base model to mitigate any potential bias between the tagging model and the experimental model. We use open-instruct[3] and lm-eval (Gao et al., 2024a) for all tests.

Baselines We conduct the following experiments for comprehensive comparison:

- **Base**: We evaluate the pre-trained Llama2-7B-Base model on the benchmarks.
- **ALL**: We train the Llama2-7B-Base model using all the data from the data pool.
- **Random**: We randomly sample data from the data pool to train the Llama2-7B base model.
- **INSTAG**: **(1)** For the diversity-driven general scenario, we adopt the approach outlined by Lu et al. (2024), utilizing their released data annotation model to label the training data. INSTAG's sampling method for ensuring diversity differs from ours and involves two steps. First, data is selected from the pool to increase the variety of chosen tags, continuing until the proportion

[1] https://huggingface.co/Qwen/Qwen2.5-7B
[2] https://huggingface.co/meta-llama/Llama-2-7b
[3] https://github.com/allenai/open-instruct

Methods	ARC-C	MMLU	BBH	C-EVAL	AVG.
Baselines					
Base	43.5	45.2	**41.6**	31.9	40.6
All	44.5	45.9	39.6	35.6	41.4
Random	45.0	45.5	39.8	32.9	40.8
InsTag	44.8	45.8	39.3	33.2	40.8
Our Methods					
CDT_Cognition	45.4	45.7	38.1	34.3	40.9
CDT_Domain	**46.3**	46.1	39.1	35.6	41.8
CDT_Task	45.4	46.2	38.8	34.2	41.2
CDT	45.5	**46.8**	40.2	**36.4**	**42.2**

Table 2: Results of applying CDT in diversity-driven general data selection, using 20% of the data pool for training. **Bold** indicating the best performance and underline indicating the second-best performance.

of selected tags relative to the total tag count reaches a threshold r. Second, data is randomly sampled from the tags that have already been selected. In this work, we set $r = 1$ based on INSTAG's findings. **(2)** For the capability-oriented specific scenario, we use only the INSTAG annotator for tag labeling. We then average the sample data from the data pool based on the capabilities tagged in the valid set.

Configuration We fine-tune the Llama2-7B-Base model using Low-Rank Adaptation (LoRA) (Hu et al., 2022), specifically targeting the attention module. Distributed training is conducted using DeepSpeed (Rasley et al., 2020). During training, the maximum sequence length is set to 2048, with a batch size of 64 and training epochs as 3.

5.2 Experiments in the General Scenario

We begin by discussing the application of the CDT in data selection for the diversity-driven general scenario. Using CDT, we can easily obtain the capability distribution of data within the pool. Given the importance of data diversity in the general scenario, we conduct experiments using the capability diversity selection method we define in Section 4.1.

Benchmarks To validate and apply our proposed capability framework in the diversity-driven general scenario, we conduct experiments using the following benchmarks: **ARC-C** (Clark et al., 2018): The ARC (AI2 Reasoning Challenge) dataset contains multiple-choice questions, focusing on science questions from grades 3 to 9. We use the Challenge portion of the dataset for testing, with accuracy as the evaluation metric. **MMLU** (Hendrycks et al., 2021): A general benchmark designed to assess knowledge acquired during pretraining by evaluating models in zero-shot and few-shot settings across several tasks. We report the average accuracy of our models under 5-shot settings. **BBH** (Srivastava et al., 2022): This benchmark includes a variety of challenging tasks with over 200 sub-tasks, many of which require higher-order and multi-step reasoning. We use the CoT prompt for testing and evaluate performance using accuracy as the metric. **C-Eval** (Huang et al., 2023): A foundational model evaluation framework in Chinese, encompassing multiple-choice tasks across domains such as STEM, Humanities, and more. We use accuracy on 5-shot as the evaluation metric.

Volume	Methods	ARC-C	BBH	MMLU	C-EVAL	AVG.
5%	**INSTAG**	44.3	38.3	44.4	32.1	39.8
	CDT	**45.7**	39.4	46.2	33.4	41.2
20%	**INSTAG**	44.8	39.3	45.8	33.2	40.8
	CDT	45.5	**40.2**	**46.8**	**36.4**	**42.2**
40%	**INSTAG**	45.2	39.4	46.3	33.7	41.2
	CDT	45.5	39.5	46.3	35.5	41.7

Table 3: The results of our method across different data selection volumes and our approach achieve the optimal results at 20%. The results are presented with **bold** indicating the best performance and underline indicating the second-best performance.

Results As shown in Table 2, our method achieves the best overall performance in the diversity-driven general scenario, with a score of 42.2. This represents 1.6 points improvement over the base model and 1.4 points improvement over the Random and INSTAG methods. Across the four benchmarks, we achieve the best results on MMLU and C-Eval, and second-best results on BBH and ARC-C. At the same time, since our method considers the data's corresponding capabilities from three dimensions, we also conduct separate experiments for each dimension. Notably, even when considering a single capability dimension, our method outperforms both the Random and INSTAG methods. Among these, the method considering only the domain dimension achieves the best results on ARC-C, while overall, it ranks second-best. These results highlight the accuracy of our CDT framework in defining capabilities and demonstrate the effectiveness of our approach for data selection in the diversity-driven general scenario.

Impact of Data Volume on CDT Performance We conduct experiments by selecting 5%, 20%, and 40% of the data from the overall data pool. The results are presented in Table 3. Using 20%

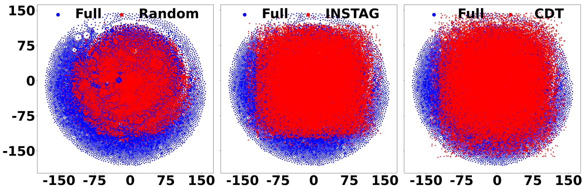


Figure 2: Diversity analysis using t-SNE on the data selected by Random, Instag, and CDT method.

of the data, our method, CDT, yields the best performance, achieving improvements of 1.0 and 0.5 compared to the 5% and 30% data selections, respectively. However, even at these data volumes, our CDT data selection methods still outperform INSTAG in all cases. These results highlight the robustness and stability of our approach across different data volumes. Based on these findings, we chose to use the 20% data configuration for the remaining experiments.

Comparison of Data Diversity Across Methods In the diversity-driven general scenario, the diversity of training data is crucial for model performance. To further validate the effectiveness of the CDT method, we conduct a diversity analysis of the data selected using the Random, INSTAG, and CDT methods from the data pool. Following the approach in (Gao et al., 2024b), we use the Llama2-7B-Chat model to extract data representations and apply t-SNE for dimensionality reduction. As shown in Figure 2, the CDT method demonstrates greater diversity compared to both Random and INSTAG. This advantage in data diversity aligns with the performance improvements observed in our benchmark tests, explaining why CDT outperforms other methods in the diversity-driven general scenario. It further reinforces the rationale behind the capability definitions in our CDT framework.

5.3 Experiments in Specific Scenario

Our data selection method has demonstrated excellent performance in the diversity-driven general scenario. However, the performance of CDT in the capability-oriented specific scenario still requires further experimental validation. For capability-oriented specific scenario, models require data with certain capabilities tailored to particular needs. We conduct experiments using the data selection method designed in Section 4.2. In this case, we select three relevant test datasets, each representing a specific capability dimension.

Methods	$\mathcal{C}$: **Concept Abstraction**	$\mathcal{D}$: **History**	$\mathcal{T}$: **Reading Comprehension**		**AVG.**
	Acc.	**Acc.**	**EM**	**F1**	
Base	0.2	51.0	0.0	2.4	13.4
All	7.5	51.3	76.0	84.9	54.9
Random	16.9	51.2	67.9	78.5	53.6
InsTag	27.3	52.4	70.5	80.7	57.7
CDT	**27.7**	**55.0**	**76.3**	**85.8**	**61.2**

Table 4: The results of using CDT for data selection in the capability-oriented specific scenario. Our method achieves the highest performance across tests in all three dimensions. $\mathcal{C}$ represents the cognition; $\mathcal{D}$ represents the domain; and $\mathcal{T}$ represents the task.

Test Datasets For each capability dimension, we select corresponding tasks for testing. We conduct experiments using the following test datasets:

- $\mathcal{C}$: In the Cognition dimension, we select the **MedQA** dataset for testing. It is a medical-related multiple-choice question dataset, and we conduct experiments on its English subset, which includes 1,273 test samples and 1,272 validation set samples. The capabilities required in the cognition dimension for MedQA are primarily HP (Hypothesis Generation) and CA (Concept Abstraction). We use accuracy (Acc.) as the evaluation metric;

- $\mathcal{D}$: For the domain dimension, we re-sample four **history**-related tasks from the MMLU benchmark, creating a multiple-choice test set focused on world history and European history, which includes 930 test samples and 121 validation set samples. In the Domain dimension, it primarily extracts History data for training. We also use accuracy as the evaluation metric.

- $\mathcal{T}$: We choose **SQuAD** (Rajpurkar et al., 2016) as the test task for the task dimension, as it primarily requires Closed Book QA and Extractive QA abilities. It is a question-answering task collected from Wikipedia, containing a 10.6k test set and no validation set, where answers must be retrieved from the provided material. We randomly split 200 samples from the test set to form a validation set. We use Exact Match (EM) and F1 score as metrics.

To align with the application method proposed in Section 4.2, we select a maximum of 200 samples from the validation set of each task for tagging and data selection. For datasets that do not contain enough samples, we use the full validation set.

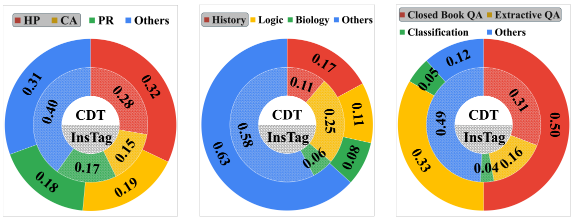
Result As shown in Table 4, our method outperforms all others across three test datasets, achieving significant improvements: 47.8 points higher than Llama2-7B-Base, 6.6 points higher than the fine-tuned Llama2-7B-Base, and 7.6 and 3.5 points higher than the Random and INSTAG methods, respectively. When using all data for training, performance on the $\mathcal{T}$ test set is nearly as good as our method, with only 0.3 points lower score in EM and 0.9 points lower in F1 compared to CDT. However, given the potential issue of imbalanced data distribution in the data pool, fine-tuning on all data for the $\mathcal{C}$ test results in even lower scores than Random. These results highlight the exceptional performance of our method in capability-oriented specific scenario, demonstrating its effectiveness.

Reasonability of Selected Data To further explore the differences between our method and INSTAG, we analyze the distribution of capability dimensions by comparing the data selected by the INSTAG method with the tags annotated by CDT. All distributions are presented in Figure 3.

In the $\mathcal{C}$ test, MedQA, we analyze the data distributions selected along the cognition dimension by INSTAG and CDT, as shown in Figure 3a. The distribution indicates that both CDT and INSTAG maintain a high degree of consistency, selecting more data from the HP and CA capabilities. However, the CDT-guided capability extraction method selects approximately 10% more data for the corresponding capabilities compared to INSTAG. This aligns with the test results, where our score is 0.4 higher than that of INSTAG, demonstrating the superior performance of our CDT-guided approach in capturing cognition dimension capabilities and improving data selection.

In the $\mathcal{D}$ test data, the History subset of MMLU, as shown in Figure Figure 3b, we observe that our CDT method prioritizes selecting History-related data to enhance Historical capability while also incorporating Logic capability to strengthen the model's reasoning ability. In contrast, the INSTAG method, although it identifies both History and Logic capabilities, confuses the relationship between the two. INSTAG selects twice as much data for Logic capabilities as for History, resulting in 2.6 points lower test score compared to the CDT method. This highlights the effectiveness of our CDT method in accurately identifying capabilities and avoiding misjudgments in related capabilities.

For the $\mathcal{T}$ tests, SQuAD, we focus the analysis on the task dimension, as shown in Figure 3c. The CDT capability distribution shows that it accurately focuses on the Closed Book QA and Extractive QA capabilities, with a highly concentrated selection of data. In contrast, although INSTAG also identifies these two capabilities, only 47% of the data selected falls within the corresponding capability range, a significant gap compared to the 83% achieved by the CDT method. This discrepancy leads to INSTAG performing substantially worse than CDT on the test data, with 5.8 points lower in EM and 5.1 points lower in F1. These findings further validate the correctness and rationality of our capability framework and highlight the exceptional performance of our method.



(a) Cognition (b) Domain (c) Task

Figure 3: The comparison of capability distribution for selected test data between CDT and INSTAG is shown. The gray areas in the figure represent the capabilities required by each task.

6 Conclusion

In this work, we introduce the Cognition-Domain-Task (CDT) capability framework, offering a comprehensive and systematic approach to classify and decompose the capabilities of LLMs. By defining cognitive abilities based on Cattell-Horn-Carroll (CHC) theory and organizing domain and task capabilities into a structured taxonomy, we enable more nuanced categorization of LLM capabilities across various scenarios. Additionally, we trained a high-quality annotator on the Qwen2.5 model using the CDT framework.

We also propose diversity-driven general data selection and capability-oriented specific data selection methods to further leverage the CDT framework. Through experiments on multiple benchmarks and test sets, we validate the correctness and stability of the CDT framework. In both scenarios, the data selection process results in significant improvements in model performance. We will release the CDT framework's construction code and model to the community to support further research.

Limitations

Our method constructs a detailed three-dimensional LLM capability framework, CDT, and explores its application in two directions: the diversity-driven general scenario and the capability-oriented specific scenario. We demonstrate improvements on the Llama2-7B-Base model. However, there are still some limitations.

First, although the annotator trained on the Qwen-2.5 model achieves higher labeling accuracy across the three dimensions compared to INSTAG, there is still significant room for improvement. This could be addressed by adding more training data or incorporating specific knowledge from human experts to guide more accurate annotator training.

Second, when defining the three dimensions, we filter out multimodal capabilities, limiting the applicability of the CDT framework to a broader range of multimodal models. Future research could expand CDT to include relevant multimodal capability classifications and conduct experiments on multimodal models such as Qwen-VL (Bai et al., 2023) and Llama-3.2 (Grattafiori et al., 2024). So far, we have focused on experiments with the Llama2-7B model to validate the capabilities of the CDT framework. Future work could extend these experiments to other models (Team, 2024; Grattafiori et al., 2024), and explore the application of CDT beyond LLMs by expanding its capabilities to MLLMs (Ye et al., 2024a; Liu et al., 2025).

Lastly, in our application of the CDT framework to LLMs, we have only explored its data selection methods across different scenarios. Future research may benefit from combining curriculum learning methods, such as Regmix (Liu et al., 2024), with the CDT framework to dynamically adjust data distribution during training, potentially leading to even better results.

References

Stephen Bach, Victor Sanh, Zheng Xin Yong, Albert Webson, Colin Raffel, Nihal V. Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, et al. 2022. PromptSource: An integrated development environment and repository for natural language prompts. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 93–104, Dublin, Ireland. Association for Computational Linguistics.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *arXiv preprint arXiv:2308.12966*.

John B Carroll. 2003. The higher-stratum structure of cognitive abilities: Current evidence supports g and about ten broad factors. *The scientific study of general intelligence*, pages 5–21.

John Bissell Carroll. 1993. *Human cognitive abilities: A survey of factor-analytic studies*. Cambridge University Press.

Raymond B Cattell. 1963. Theory of fluid and crystallized intelligence: A critical experiment. *Journal of educational psychology*, 54(1):1.

Mayee F. Chen, Nicholas Roberts, Kush Bhatia, Jue Wang, Ce Zhang, Frederic Sala, and Christopher Ré. 2023. Skill-it! A data-driven skills framework for understanding and training language models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free dolly: Introducing the world's first truly open instruction-tuned llm.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*.

Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. 2023. Enhancing chat language models by scaling high-quality instructional conversations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3029–3051, Singapore. Association for Computational Linguistics.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. 2024. Length-controlled alpacaeval: A simple way to debias automatic evaluators. *arXiv preprint arXiv:2404.04475*.

Dawn P Flanagan, Kevin S McGrew, and Samuel O Ortiz. 2000. *The Wechsler Intelligence Scales and Gf-Gc theory: A contemporary approach to interpretation*. Allyn & Bacon.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2024a. A framework for few-shot language model evaluation.

Pengzhi Gao, Zhongjun He, Hua Wu, and Haifeng Wang. 2024b. Towards boosting many-to-many multilingual machine translation with large language models. *CoRR*, abs/2401.05861.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

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

John Leonard Horn. 1965. *Fluid and crystallized intelligence: A factor analytic study of the structure among primary mental abilities*. University of Illinois at Urbana-Champaign.

Bin Hu, Chenyang Zhao, Pu Zhang, Zihao Zhou, Yuanhang Yang, Zenglin Xu, and Bin Liu. 2023. Enabling intelligent interactions between an agent and an llm: A reinforcement learning approach. *CoRR*, abs/2306.03604.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Jiayi Lei, et al. 2023. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, et al. 2023. Openassistant conversations - democratizing large language model alignment. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Aviral Kumar, Vincent Zhuang, Rishabh Agarwal, Yi Su, John D Co-Reyes, Avi Singh, Kate Baumli, Shariq Iqbal, Colton Bishop, Rebecca Roelofs, et al. 2024. Training language models to self-correct via reinforcement learning. *arXiv preprint arXiv:2409.12917*.

Young-Jun Lee, Dokyong Lee, Junyoung Youn, Kyeongjin Oh, and Ho-Jin Choi. 2024. Thanos: Enhancing conversational agents with skill-of-mind-infused large language model. *arXiv preprint arXiv:2411.04496*.

Antonio Lieto, Mehul Bhatt, Alessandro Oltramari, and David Vernon. 2018. The role of cognitive architectures in general artificial intelligence.

Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. 2024. Regmix: Data mixture as regression for language model pre-training. *CoRR*, abs/2407.01492.

Zuyan Liu, Yuhao Dong, Ziwei Liu, Winston Hu, Jiwen Lu, and Yongming Rao. 2025. Oryx MLLM: On-demand spatial-temporal understanding at arbitrary resolution. In *The Thirteenth International Conference on Learning Representations*.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, et al. 2023. The flan collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 22631–22648. PMLR.

Keming Lu, Hongyi Yuan, Zheng Yuan, Runji Lin, Junyang Lin, Chuanqi Tan, Chang Zhou, and Jingren Zhou. 2024. #instag: Instruction tagging for analyzing supervised fine-tuning of large language models. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Kevin S McGrew and Jeffrey J Evans. 2004. Internal and external factorial extensions to the cattell-horn-carroll (chc) theory of cognitive abilities: A review of factor analytic research since carroll's seminal 1993 treatise. *Institute for Applied Psychometrics*.

Brando Miranda, Alycia Lee, Sudharsan Sundar, Allison Casasola, Rylan Schaeffer, and Sanmi Koyejo. 2024. Beyond scale: The diversity coefficient as a data quality metric for variability in natural language data. In *ICLR 2024 Workshop on Data-centric Machine Learning Research (DMLR): Harnessing Momentum for Science*. OpenReview.net.

Kolby Nottingham, Bodhisattwa Prasad Majumder, Bhavana Dalvi Mishra, Sameer Singh, Peter Clark, and Roy Fox. 2024. Skill set optimization: Reinforcing language model behavior via transferable skills.

In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

OpenAI. 2024a. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

OpenAI. 2024b. Learning to reason with llms.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Felipe Maia Polo, Seamus Somerstep, Leshem Choshen, Yuekai Sun, and Mikhail Yurochkin. 2025. Sloth: scaling laws for LLM skills to predict multi-benchmark performance across families.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. 2020. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 3505–3506. ACM.

W Joel Schneider and Kevin S McGrew. 2018. The cattell-horn-carroll theory of cognitive abilities. *Contemporary intellectual assessment: Theories, tests, and issues*, pages 73–163.

Wei Song, Yadong Li, Jianhua Xu, Guowei Wu, Lingfeng Ming, Kexin Yi, Weihua Luo, Houyi Li, Yi Du, Fangda Guo, et al. 2024. M3gia: A cognition inspired multilingual and multimodal general intelligence ability benchmark. *arXiv preprint arXiv:2406.05343*.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *CoRR*, abs/2206.04615.

Qwen Team. 2024. Qwen2.5: A party of foundation models.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. 2024a. Math-shepherd: Verify and reinforce LLMs step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9426–9439, Bangkok, Thailand. Association for Computational Linguistics.

Xiaoqiang Wang, Lingfei Wu, Tengfei Ma, and Bang Liu. 2024b. FAC^{2}E: Better understanding large language model capabilities by dissociating language and cognition. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13228–13243, Miami, Florida, USA. Association for Computational Linguistics.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023a. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023b. 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.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, et al. 2022. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Zhihu Wang, Shiwan Zhao, Yu Wang, Heyuan Huang, Sitao Xie, Yubo Zhang, Jiaxin Shi, Zhixing Wang, Hongyan Li, and Junchi Yan. 2024c. Re-task: Revisiting llm tasks from capability, skill, and knowledge perspectives. *CoRR*, abs/2408.06904.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Minghao Wu, Thuy-Trang Vu, Lizhen Qu, and Reza Haf. 2024. Mixture-of-skills: Learning to optimize data usage for fine-tuning large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 14226–14240, Miami, Florida, USA. Association for Computational Linguistics.

Congying Xia, Chen Xing, Jiangshu Du, Xinyi Yang, Yihao Feng, Ran Xu, Wenpeng Yin, and Caiming

Xiong. 2024. FOFO: A benchmark to evaluate LLMs' format-following capability. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 680–699, Bangkok, Thailand. Association for Computational Linguistics.

Zifan Xu, Haozhu Wang, Dmitriy Bespalov, Xuan Wang, Peter Stone, and Yanjun Qi. 2023. Latent skill discovery for chain-of-thought reasoning. *arXiv preprint arXiv:2312.04684*.

Jiabo Ye, Haiyang Xu, Haowei Liu, Anwen Hu, Ming Yan, Qi Qian, Ji Zhang, Fei Huang, and Jingren Zhou. 2024a. mplug-owl3: Towards long image-sequence understanding in multi-modal large language models. *CoRR*, abs/2408.04840.

Seonghyeon Ye, Doyoung Kim, Sungdong Kim, Hyeonbin Hwang, Seungone Kim, Yongrae Jo, James Thorne, Juho Kim, and Minjoon Seo. 2024b. FLASK: fine-grained language model evaluation based on alignment skill sets. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Jian Zhao, Mengqing Wu, Liyun Zhou, Xuezhu Wang, and Jian Jia. 2022. Cognitive psychology-based artificial intelligence review. *Frontiers in Neuroscience*, 16:1024316.

Ming Zhong, Aston Zhang, Xuewei Wang, Rui Hou, Wenhan Xiong, Chenguang Zhu, Zhengxing Chen, Liang Tan, Chloe Bi, Mike Lewis, et al. 2025. Law of the weakest link: Cross capabilities of large language models. In *The Thirteenth International Conference on Learning Representations*.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, LILI YU, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. Lima: Less is more for alignment. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

A Appendix

A.1 Prompt

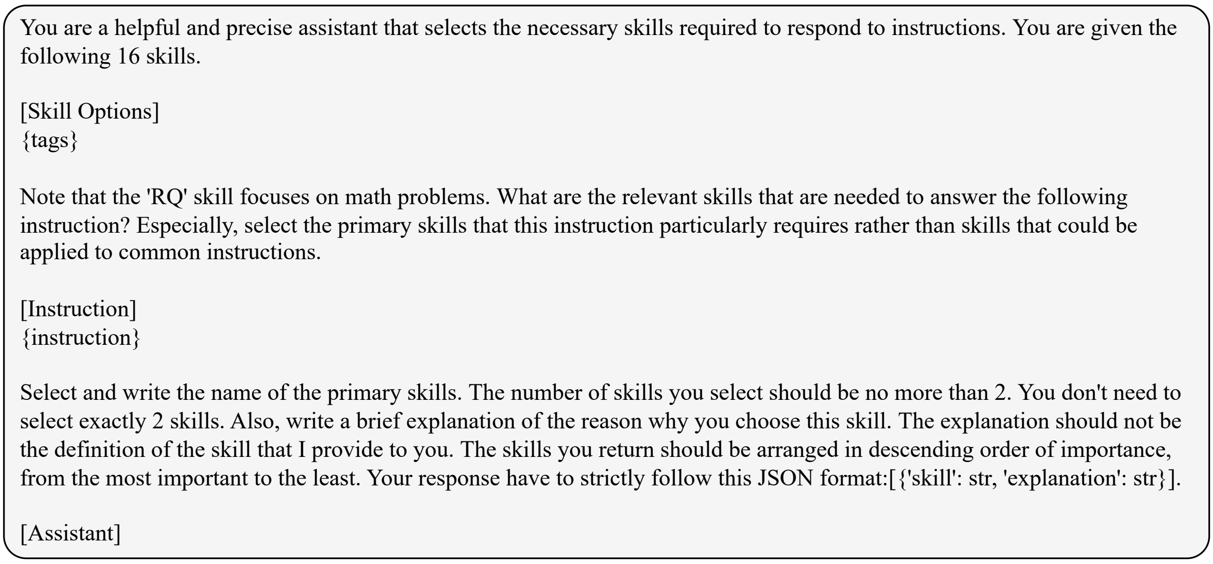
The prompts used for training the annotator and labeling data with the annotator are shown in Figure 4. We concatenate the detailed descriptions of the query, tag, and instruction into a single input prompt. When labeling the cognition dimension, we restrict the model to output at most two tags, along with their corresponding explanations.

A.2 Capability Definition

The detailed definitions and abbreviations for the cognition, domain, and task dimensions are provided in Table 5, Table 6, and Table 7, respectively. In defining the domain dimension, we first established the overarching domain and then carefully subdivided it into subdomains for labeling purposes.

A.3 Data Selection Algorithm

We present our diversity-driven general scenario data selection algorithm in Algorithm 1 and capability-oriented specific scenario in Algorithm 2.



You are a helpful and precise assistant that selects the necessary skills required to respond to instructions. You are given the following 16 skills.

[Skill Options]
{tags}

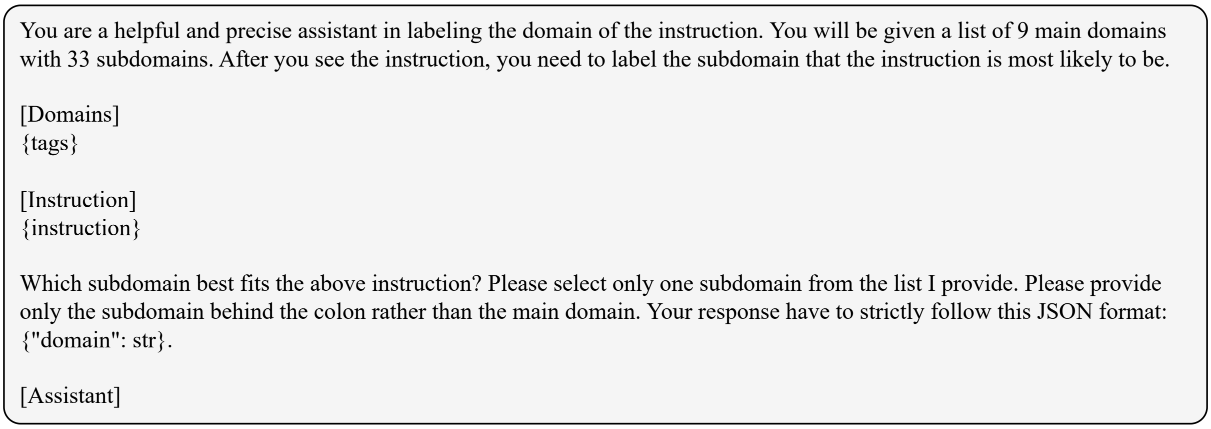
Note that the 'RQ' skill focuses on math problems. What are the relevant skills that are needed to answer the following instruction? Especially, select the primary skills that this instruction particularly requires rather than skills that could be applied to common instructions.

[Instruction]
{instruction}

Select and write the name of the primary skills. The number of skills you select should be no more than 2. You don't need to select exactly 2 skills. Also, write a brief explanation of the reason why you choose this skill. The explanation should not be the definition of the skill that I provide to you. The skills you return should be arranged in descending order of importance, from the most important to the least. Your response have to strictly follow this JSON format:[{'skill': str, 'explanation': str}].

[Assistant]

(a) Cognition tagging prompt



You are a helpful and precise assistant in labeling the domain of the instruction. You will be given a list of 9 main domains with 33 subdomains. After you see the instruction, you need to label the subdomain that the instruction is most likely to be.

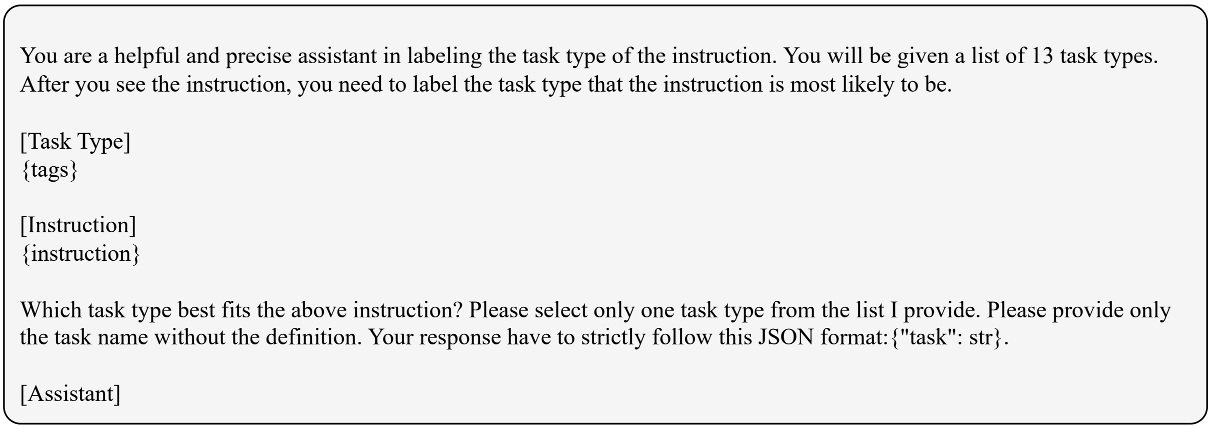
[Domains]
{tags}

[Instruction]
{instruction}

Which subdomain best fits the above instruction? Please select only one subdomain from the list I provide. Please provide only the subdomain behind the colon rather than the main domain. Your response have to strictly follow this JSON format: {"domain": str}.

[Assistant]

(b) Domain tagging prompt



You are a helpful and precise assistant in labeling the task type of the instruction. You will be given a list of 13 task types. After you see the instruction, you need to label the task type that the instruction is most likely to be.

[Task Type]
{tags}

[Instruction]
{instruction}

Which task type best fits the above instruction? Please select only one task type from the list I provide. Please provide only the task name without the definition. Your response have to strictly follow this JSON format:{"task": str}.

[Assistant]

(c) Task tagging prompt

Figure 4: The prompts we used on tagging.

Cognition	**Abbreviation**	**Definition**
Pattern Recognition	PR	Ability to identify recurring patterns, trends, or sequences within a given set of data or materials (e.g., detecting similarities in a sequence of numbers or text).
Concept Abstraction	CA	Ability to form abstract concepts or categories based on shared characteristics or relationships among a set of materials.
Hypothesis Generation	HP	Ability to propose plausible explanations or predictions for incomplete information (e.g., inferring causes of a fictional conflict, suggesting scientific hypotheses).
General Sequential Reasoning	RG	Ability to start with stated rules, premises, or conditions, and to engage in one or more steps to reach a solution to a novel problem.
Quantitative Reasoning	RQ	Ability to inductively and deductively reason with concepts involving mathematical relations and properties.
Communication Ability	CM	Ability to mimic speak in real-life situations (e.g., lecture, group participation) in an adult-like manner.
Mathematical Achievement	A3	Measured mathematics achievement.
Reading Decoding	RD	Ability to recognize and decode words or pseudowords in reading.
Writing Ability	WA	Ability to write with clarity of thought, organization, and good sentence structure.
Naming Facility	NA	Ability to rapidly produce names for concepts when presented with a text cue.
Associational Fluency	FA	Ability to rapidly produce a series of original or useful ideas related to a particular concept.
Expressional Fluency	FE	Ability to rapidly think of different ways of expressing an idea.
Sensitivity to Problems/Alternative Solution Fluency	SP	Ability to rapidly think of a number of solutions to particular practical problem.
Originality/ Creativity	FO	Ability to rapidly produce original, clever, and insightful responses (expressions, interpretations) to a given topic, situation, or task.
Ideational Fluency	FI	Ability to rapidly produce a series of ideas, words, or phrases related to a specific condition or object. Quantity, not quality, is emphasized.
Word Fluency	FW	Ability to rapidly produce words that have specific phonemic, structural, or orthographic characteristics (independent of word meanings).

Table 5: The full definition of Cognition.

Domain	**Sub-domain**
Language	Linguistics,Literature,Multilingualism
Culture	Tradition,Art,Sports,Mass Media,Music,Food
Health	Health
Natural Science	Biology,Earth Science,Astronomy,Chemistry,Physics
Math	Mathematics,Logic
Social Science	Economics,Law,Politics,Education,Sociology
Technology	Agriculture,Computer Science,Automation,Electronics,Engineering
Coding	Coding
Humanities	Communication,Religion,Philosophy,Ethics,History

Table 6: The full definition of Domain.

Task	Definition
Generation	Creating new information with human-input conditions, involving the automatic generation of various text materials follow the instruction given by the user.
Rewrite	Taking a piece of text and rephrasing it while preserving its original meaning, which may involve simplifying the language, changing the structure, or adjusting the tone.
Summarization	Condensing longer texts into shorter versions while retaining the key information and main ideas, making it easier to digest complex information.
Classification	Assigning predefined labels or categories to text based on its content, such as topic categorization.
Brainstorming	Generating ideas, encouraging creative thinking, or exploring possibilities.
Sentiment	Determining the emotional tone or sentiment expressed in a piece of text.
Completion	Continuing a given prompt with relevant and contextually appropriate content, such as finishing sentences or filling in blanks.
Natural Language Inference	Assessing the relationship between two sentences to determine if one logically follows from the other (entailment), (contradiction), or if the relationship is unclear (neutral).
Bias and Fairness	Evaluating models for potential bias, fairness, or harmfulness in their outputs.
Word Sense Disambiguation	Determining which meaning of a word is used in a given context, especially for words that have multiple meanings.
Multiple Choice QA	Answering questions by selecting the correct option from a predefined set of possible answers based on provided information or context.
Closed Book QA	Answering questions directly without access to external knowledge.
Extractive QA	Identifying and extracting specific pieces of information from a given text to answer the question.

Table 7: The full definition of Task.

Algorithm 1: Diversity-driven General Scenario Data Selection

```
Data: $D^{'}_{pool}$: The capacity labeled data pool; $N$: Selection set size;
Result: $D_{train}$: The selected training dataset;
initialization: $T_d$: All composite capabilities in the data pool; $D_{train} \leftarrow \emptyset$;
Sorting $T_d$ in descending order based on the number of corresponding data points in $D^{'}_{pool}$;
while $|D_{train}| < N$ do
    $Flag \leftarrow False$;
    for each capability $f \in T_d$ do
        $D_f \leftarrow Find\_Data(f, D^{'}_{pool})$;
        // Selecting data tagged with composite capability $f$ from $D^{'}_{pool}$
        if $D_f \neq \emptyset$ then
            $d \leftarrow Random(D_f, 1)$;
            // Selecting one data point randomly from $D_f$
            $D_{train} \leftarrow \{d\} \cup D_{train}$;
            $D^{'}_{pool} \leftarrow D^{'}_{pool} \backslash \{d\}$;
            $Flag \leftarrow True$;
        end
        if $|D_{train}| = N$ then
            break;
        end
    end
    if $Flag = False$ then
        break;
        // All data points related to capability set $T_d$ are selected
    end
end
```

Algorithm 2: Capability-oriented Specific Scenario Data Selection

```
Data: $D^{'}_{pool}$: The capacity labeled data pool; $D^{'}_{valid}$: The capacity labeled validation set; $N$: Selection set size;
Result: $D_{train}$: The selected training dataset;
initialization: $T_v$: Triplet capability set of validation set; $T_v^{*}$: Binary capability set; $T_v^{\star}$: Unary capability set; $D_{train} \leftarrow \emptyset$;
for each capability set $T \in \{T_v, T_v^{*}, T_v^{\star}\}$ do
    Sorting $T$ in descending order based on the number of corresponding data points in $D^{'}_{pool}$;
    while $|D_{train}| < N$ do
        $Flag \leftarrow False$;
        for each capability $f \in T$ do
            if $N = |D_{train}|$ then
                break;
            end
            $D_f \leftarrow Find\_Data(f, D^{'}_{pool})$;
            // Selecting data tagged with composite capability $f$ from $D^{'}_{pool}$
            if $D_f \neq \emptyset$ then
                $d \leftarrow Random(D_f, 1)$;
                // Selecting one data point randomly from $D_f$
                $D_{train} \leftarrow \{d\} \cup D_{train}$;
                $D^{'}_{pool} \leftarrow D^{'}_{pool} \backslash \{d\}$;
                $Flag \leftarrow True$;
            end
        end
        if $Flag = False$ then
            break;
            // All data points related to capability set $T$ are selected
        end
    end
end
if $|D_{train}| < N$ then
    // Not enough data points labeled with the desired capabilities
    $D_r \leftarrow Random(D^{'}_{pool}, N - |D_{train}|)$;
    $D_{train} \leftarrow D_r \cup D_{train}$;
end
```