Probing and Boosting Large Language Models Capabilities via Attention Heads

Dezhi Zhao[1,2], **Xin Liu**[2,*], **Xiaocheng Feng**[1,2,*], **Hui Wang**[2], **Bing Qin**[1,2]

[1] Harbin Institute of Technology, [2] Peng Cheng Laboratory
{dzzhao, xcfeng, qinb}@ir.hit.edu.cn,
hit.liuxin@gmail.com, wangh06@pcl.ac.cn

Abstract

Understanding the internal origins of capabilities in large language models (LLMs) is crucial for interpretability and efficient adaptation. However, the emergence of specific capabilities remains poorly understood, as most existing approaches rely on external signals (e.g., performance shifts or gradient similarities) with limited structural grounding. To address these issues, this paper proposes a lightweight and highly interpretable approach that links LLM capabilities to internal components by identifying correspondences at the level of attention heads. Specifically, we first define five fundamental capabilities, namely Mathematical Reasoning, Reading Comprehension, Commonsense Reasoning, Scientific Reasoning, and Professional Expertise, and employ probing techniques to detect the attention heads most predictive of each, thereby establishing capability–head mappings. For targeted instruction tuning, complex tasks are decomposed into these fundamental capabilities, and training data are selected accordingly. Experiments on LLaMA3.1-8B and Qwen2.5-7B show over 70% discrimination accuracy in identifying capabilities. On MMLU and BBH, our method improves accuracy by 1 to 1.5 points over the gradient-based method LESS and by 5 to 6 points over other intermediate-state baselines[1].

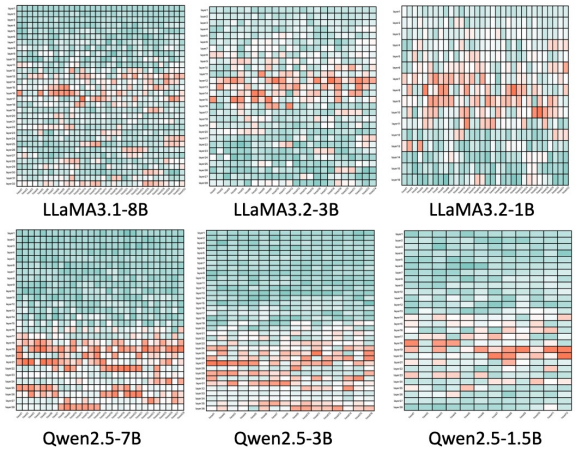

Figure 1: Internal distribution of reading comprehension capability within the LLaMA3 and Qwen2.5 model series. Each cell represents an attention head, where increasing color intensity (towards red) denotes higher accuracy on reading comprehension question answering. The visualization indicates that models of different scales within the same series exhibit similar patterns in their internal capability distributions.

1 Introduction

Large language models (LLMs) (Grattafiori et al., 2024; Achiam et al., 2023; Team et al., 2024; Xu et al., 2025) have achieved remarkable success across a wide range of natural language processing and cross-modal tasks, largely driven by scaling laws and the availability of massive training data (Kaplan et al., 2020; Hoffmann et al., 2022; Wei et al., 2022a). As models continue to grow in size and capability, they demonstrate increasingly sophisticated behaviors such as multi-step reasoning, instruction following, and domain-specific expertise. Nevertheless, a precise understanding of how specific data segments contribute to targeted LLM capabilities remains elusive.

Although prior work has investigated methods for selecting high-quality data to improve LLM capabilities (Xia et al., 2024; Schioppa et al., 2022; Zhou et al., 2023), these approaches generally quantify the contribution of data points to model learning or final performance. Such analyses are largely phenomenological, describing correlations at the data–performance level without establishing a principled link between training data, emergent capabilities, and internal model components, thereby limiting fine-grained interpretability.

Insights into internal model components offer a promising direction (Li et al., 2024; Liang et al.,

*Corresponding author
[1]https://github.com/Dezhi93/ProBooCap

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 28519–28533
November 4-9, 2025

Capability	Representative Dataset	Instances Num	Answer Type	Abbreviation
Mathematical Reasoning	MathQA	5970	COT choice	MR
Reading Comprehension	Race	6996	Choice	RC
Commonsense Reasoning	TruthfulQA	5918	Generation	CR
Scientific Reasoning	ScienceQA	4448	Choice	SR
Professional Expertise	MedQA&LegalQA	5546	Choice	PE

Table 1: Statistics for defined fundamental capabilities and their corresponding capability identification dataset. Instances Num lists the total instances per capability, subsequently split into training and validation sets (8:2 ratio). Capability abbreviations in the last column are used for reference in later figures and tables.

2024; Zhang et al., 2025). For instance, studies like ITI (Inference Time Intervention) have shown that intermediate states of attention heads can be more informative than final outputs, improving truthfulness detection by 40% (Li et al., 2024). Similarly, representations from MLP layers have been found to predict a model's inherent knowledge for a given question with over 80% accuracy (Liang et al., 2024). In parallel, researches on the internal organization of knowledge in transformers have revealed that knowledge are not diffusely stored, but are often localized within specific model components (Bills et al., 2023; Meng et al., 2022a; Dai et al., 2021; Meng et al., 2022b; Geva et al., 2023). However, this granular understanding has largely remained an analytical pursuit, seldom leveraged to directly guide and optimize the model's own training process.

Building on the premise that intermediate states harbor rich yet underexplored information, our preliminary analyses reveal consistent patterns in capability localization across model families. In the LLaMA3 series, attention heads most predictive of reading comprehension are concentrated in middle layers, whereas in the Qwen2.5 series (Yang et al., 2024) they appear in upper-middle layers (Figure 1). These observations, highlighting intra-series consistency and inter-series variation, provide concrete evidence of structure in capability distribution and motivate the hypothesis that individual attention heads correspond to distinct model capabilities.

These observations motivate us to move beyond descriptive analyses and develop a systematic method to map LLM capabilities to internal components. To this end, we propose a lightweight and interpretable approach for establishing correspondences between specific capabilities and attention heads. We first define five fundamental LLM capabilities (details in Table 1) and then identify attention heads in models such as LLaMA3.1-8B and Qwen2.5-7B that are highly predictive of each capability, providing initial evidence for the hypothesized correspondence.

Furthermore, because the primary goal of targeted instruction tuning (Xia et al., 2024) is to enhance specific capabilities, we use this setting to validate our findings. On multi-task benchmarks such as MMLU and BBH, subtasks are decomposed into combinations of the five fundamental capabilities (see Appendix A, Table 6), and capability-specific attention heads are employed to guide data selection. Our method consistently outperforms the gradient-based selection method LESS (Xia et al., 2024) and surpasses other intermediate-state-based baselines (Zhang et al., 2018; Hanawa et al., 2020), demonstrating both the practical utility and the validity of the proposed capability–head correspondence.

In summary, this work provides a new perspective on understanding and manipulating LLM capabilities through their inherent attention mechanisms. We demonstrate that a model's internal structure can be used to guide its own training, a process that fuses performance gains with model interpretability.

Our main contributions are as follows:

- We establish a systematic correspondence between fundamental LLM capabilities and internal attention heads, supported by capability-localization results that validate this relationship.
- We leverage this correspondence for targeted instruction tuning and propose a lightweight, gradient-free data selection method that is both efficient and interpretable, which provides an empirically effective alternative to conventional black-box methods.

- We show that complex capabilities in benchmarks such as MMLU and BBH can be decomposed into combinations of the defined fundamental capabilities. This composability enables the extension of our approach to unseen capabilities and highlights its potential for interpretable enhancement of LLM abilities.

2 LLMs Capability-Attention Head Correspondence

In this section, we define fundamental capabilities, describe the construction of our dataset for capability localization, introduce a probing-based approach to identify capability-specific attention heads, and conclude with an analysis of the distribution of the top 16 heads associated with different capabilities.

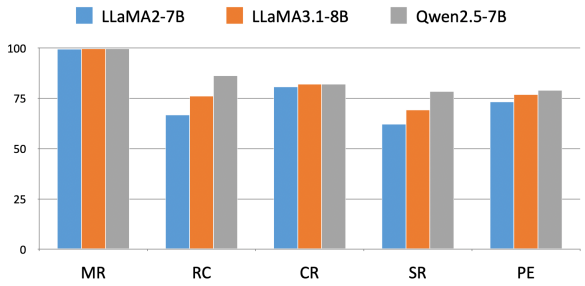

Figure 2: Per-capability classification accuracy of the top-1 attention head on 7B+ models (LLaMA2-7B, LLaMA3.1-8B, and Qwen2.5-7B).

2.1 Definition of fundamental capabilities

To investigate the correspondence between model capabilities and attention heads, we define five fundamental LLM capabilities (detailed in Appendix A, Table 8) and associate each with a distinct dataset (Table 1).

Capability-specific classification dataset construction. We construct a binary classification dataset for each capability. For most datasets, such as RACE (Lai et al., 2017), TruthfulQA (Lin et al., 2021), ScienceQA (Lu et al., 2022), MedQA (Jin et al., 2021), and LegalQA[2], positive samples consist of question-answer pairs from the development set, while negative samples are formed by pairing questions with randomly selected incorrect options. For the MathQA dataset (Amini et al., 2019), which requires Chain-of-Thought (CoT) reasoning (Wei et al., 2022b), we adopt a different strategy: negative examples are generated by GPT-4 (Achiam et al., 2023) to feature plausible yet flawed reasoning chains. Detailed statistics are provided in Table 1, and illustrative examples are presented in Appendix A (Tables 9 and 10).

[2]https://huggingface.co/datasets/bwang0911/legal_qa_v1

2.2 Localizing capability-specific attention heads with probes

Classifiers and models. Following (Li et al., 2024), we employ logistic regression classifiers as capability discriminators. Despite their simplicity, these classifiers prove highly effective, yielding high accuracy in identifying capability-specific attention heads. This effectiveness aligns with findings that weak classifiers are sufficient for probing internal model components, such as distinguishing unethical inputs in MLP layers (Zhou et al., 2024). We conduct these localization experiments across the LLaMA3 and Qwen2.5 model families, with sizes ranging from 1B to 8B parameters.

Localization results. To localize capabilities, we train a logistic regression classifier for each attention head in the model. Each classifier is trained on a specific capability dataset, using only the output representation from its corresponding attention head as input. For each capability, we identify the top-1 attention head as the one whose classifier achieves the highest accuracy on a held-out validation set. We find that for models of 7B parameters and larger, the classifiers associated with these top-1 attention heads surpass 70% accuracy on their respective capabilities (Figure 2). This strong performance indicates a clear correspondence between specific heads and high-level model capabilities.

2.3 Distribution of top-16 attention heads across capabilities

To validate the distinctiveness of our capability categorization, we visualize the top-16 attention heads for each capability (as used in Sections 3 and 4) in Figures 3 and 4. For LLaMA3.1-8B, these sets of attention heads show remarkably little overlap. As detailed in Figure 3(f), the majority of attention heads (>50%) are unique to a single capability, with only three heads shared across three capabilities. A similar pattern of low overlap is observed for LLaMA2-7B (Figure 4(f)). Together, these results demonstrate the high discriminability of our taxonomy and validate our categorization of the five fundamental capabilities.

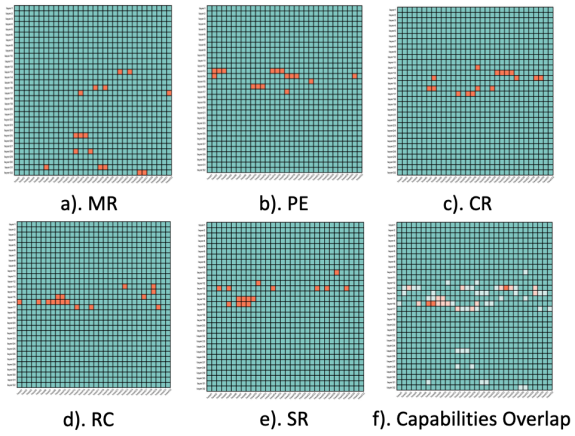

Figure 3: a)-e) Distribution of the top-16 attention heads for each of five distinct capabilities in LLaMA3.1-8B (capabilities are denoted by abbreviations). f) Degree of overlap among the top-16 attention head sets for these five capabilities; darker colors indicate an attention head is shared by a greater number of capabilities.

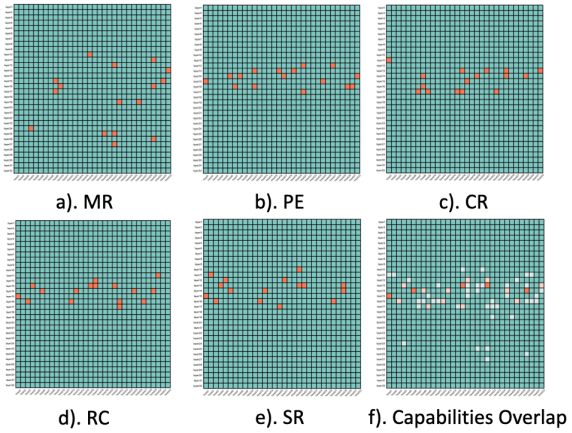

Figure 4: As in Figure 3, the model considered here is LLaMA2-7B.

3 Validation on Targeted Instruction Tuning

To further validate our capability-attention head mapping, we conduct experiments on the targeted instruction tuning task designed to enhance specific capabilities (Xia et al., 2024). Our central hypothesis is that if the mapping is accurate, leveraging the identified attention heads should improve performance on subtasks that require the corresponding capabilities. Our method involves two main steps: first, decomposing each subtask into its constituent fundamental capabilities, and second, filtering the training data using representations from the relevant attention heads. This section is organized as follows: Section 3.1 outlines our validation framework, Section 3.2 details the experimental settings, and Section 3.3 introduces the baselines.

3.1 Validation Framework

The complete validation pipeline is illustrated in Figure 5, consisting of three main modules: (1) capability-specific head localization, (2) composite capability decomposition and similarity computation, and (3) aggregated data selection (Not depicted in Figure 5; this is described separately in Algorithms 1-3).

Step 1 Capability-specific attention heads localization We first identify the attention heads corresponding to each fundamental capability, guided by the definitions in Section 2.1 and the localization methods in Section 2.2. These identified attention heads are then used for subsequent computations.

Step 2 Composite capability decomposition and similarity computation While Section 2.1 defines fundamental model capabilities, solving complex subtasks often requires combining multiple capabilities. We therefore decomposed each subtask's required capabilities (detailed in Appendix A Table 6). Upon completion of the subtask composite capability decomposition, each subtask is mapped to one or two fundamental application capabilities. Subsequently, we input few-shot examples for subtasks from the development sets of our evaluation benchmarks (MMLU and BBH) into the base models (LLaMA2-7B and LLaMA3.1-8B). This process yields intermediate representations from the attention heads corresponding to the capabilities requisite for these subtasks. These representations are considered canonical exemplars of subtask-specific capabilities and are reused in the subsequent general data selection phase.

Inspired by Dai et al. (2022), who posited that in-context learning can be an implicit form of gradient descent, we aim for high similarity between the intermediate representations of candidate data and those of in-context learning exemplars. Such alignment suggests that fine-tuning with these selected data points approximates training with near in-domain samples, thereby facilitating more effective gradient descent. Furthermore, as argued by Zhou et al. (2023); Longpre et al. (2023); Wei et al. (2021), instruction tuning primarily activates latent capabilities rather than instilling new knowledge. Therefore, selected data exhibiting greater similarity to few-shot exemplars should more effectively elicit the corresponding capabilities.

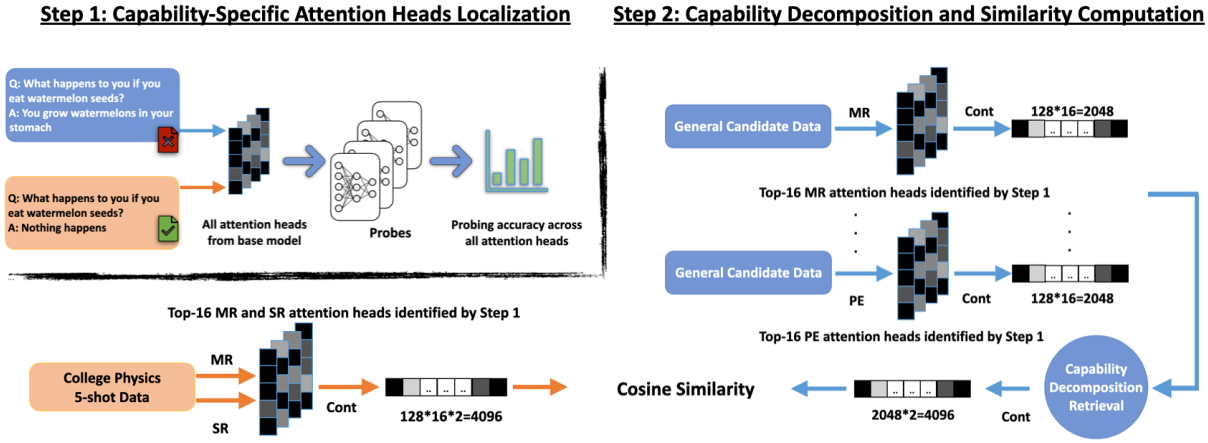

Figure 5: The validation framework employed for targeted instruction tuning task, depicting Steps 1 and 2. Step 3 is detailed independently in Algorithms 1-3. In this visualization, capabilities are abbreviated (MR, SR, PE), and Cont represents vector concatenation.

Specifically, guided by the subtask capability decomposition, we form a composite representation by concatenating the capability-specific attention heads identified in Step 1. From this composite representation, we extract the intermediate states for each candidate instance from the general data and for the few-shot representative data. The cosine similarity is then computed between the representation of a candidate instance and that of the representative data for each subtask. Finally, each candidate is assigned to the subtask with the highest similarity score, while its full set of scores across all subtasks is retained for downstream use.

The College Physics subtask within the MMLU dataset exemplifies a task requiring multiple capabilities: mathematical reasoning for calculations and scientific reasoning for inferring physical principles. To form a composite capability representation for this subtask, we concatenate the top-16 attention heads associated with each of these two capabilities, as identified in Step 1. This composite representation is subsequently used to perform similarity calculations.

Step 3 Aggregated data selection For each subtask, we aim to select an equal quantity of data for augmentation, specifically targeting a volume equivalent to Q in Algorithm 1. This selection process utilizes the capability similarity scores obtained in Step 2, where candidate general data instances are ranked in descending order of their similarity to each subtask. The detailed algorithmic procedure is illustrated in Algorithm 2.

However, the initial assignment of data instances in Step 2 (where each instance is typically assigned to the subtask with which it has the highest similarity) often leads to an imbalanced data distribution across subtasks. Consequently, some subtasks may not receive their aforementioned target quantity of data solely from instances for which they are the top-ranked similarity match.

To address this imbalance, if a subtask's target data quantity is not fulfilled by these primary assignments, we implement a supplementary selection mechanism. Candidate data instances are then considered for subtasks based on their second-highest, third-highest (and so forth) cosine similarity scores. These additionally considered instances are subsequently ranked by their respective similarity to the subtask and are selected in this rank order to augment the subtask's dataset until its predetermined data quantity is met (Algorithm 3).

3.2 Experimental Setup

We describe the experimental setup used in our analysis (Section 3 and 4).

Training(General) datasets. For comparison with LESS (Xia et al., 2024), we follow Wang et al. (2023) and utilize the same instruction tuning datasets as LESS: (1) datasets created from existing ones such as FLAN V2 (Longpre et al., 2023) and COT (Wei et al., 2022b); (2) open-ended generation datasets with human-written answers including DOLLY (Conover et al., 2023) and OPEN ASSISTANT 1 (Köpf et al., 2023).

Evaluation datasets. To compare our method with LESS, we adopt the evaluation datasets used by LESS on MMLU (Hendrycks et al., 2020) and BBH (Srivastava et al., 2022).

Method	MMLU					BBH
	STEM	Humanities	Social Sciences	Other	Avg	Avg
Rand	36.8	43.0	53.3	54.3	46.5	38.9
BM25 (Robertson et al., 2009)	37.6	44.8	54.2	54.7	47.6	39.8
DSIR (Xie et al., 2023)	36.0	43.1	53.3	53.1	46.1	36.8
RDS (Hanawa et al., 2020)	35.9	41.1	51.5	52.8	45	36.7
LESS (Xia et al., 2024)	38.9	47.6	58.7	**58.6**	50.2	41.5
Ours	**39.0**	**49.3**	**58.8**	58.1	**51.2**	**43.0**
Rand	56.0	58.2	74.9	71.3	64.4	61.9
BM25 (Robertson et al., 2009)	55.1	58.7	75.1	71.2	64.4	64.7
DSIR (Xie et al., 2023)	53.1	59.7	75.3	71.7	64.5	64.0
RDS (Hanawa et al., 2020)	54.7	59.3	74.7	70.7	64.3	64.2
LESS (Xia et al., 2024)	55.1	59.7	75.6	**72.2**	65.1	66.2
Ours	**57.0**	**61.1**	**76.0**	71.7	**65.9**	**70.2**

Table 2: Comparison of our method against LESS, RDS, DSIR, and BM25 on the LLaMA2-7B (top section) and LLaMA3.1-8B (bottom section) models, when training with the top 5% of data. Results for our method are obtained using data selected via the top-16 attention heads per capability.

Model. We evaluate our method on two base models: LLaMA2-7B and LLaMA3.1-8B. LLaMA2-7B is the model employed in the LESS, while LLaMA3.1-8B is utilized to demonstrate the effectiveness of our approach on a more recent model. Instruction tuning for all models is performed using Low-Rank Adaptation (LoRA) (Hu et al., 2022) as LESS. Training details are provided in Appendix A.1.

3.3 Baselines

In addition to our primary baseline, LESS (Xia et al., 2024), we compare our method against several baselines evaluated in the same work. These secondary baselines include BM25 (Robertson et al., 2009), DSIR (Xie et al., 2023), and RDS (Zhang et al., 2018; Hanawa et al., 2020). Among them, RDS, another data selection method that leverages a model's hidden representations, is a crucial secondary baseline for ablation, allowing us to specifically demonstrate the advantages of using capability-specific heads. For experiments on LLaMA2-7B, all baseline results are cited directly from the LESS paper. As the original work did not evaluate on LLaMA3.1-8B, we report our own reproduced results for this model.

4 Experimental Results and Analysis

In Section 4.1, we present the main experimental results. Then, in Sections 4.2 and 4.3, we respectively analyze the influence of different numbers of attention heads and varying scales of selected data on performance. Ablation studies are detailed in Section 4.4. In Section 4.5, we address the limitations of subjectivity and scalability.

4.1 Main Results

As shown in Table 2, our method achieves consistent improvements on both LLaMA2-7B and LLaMA3.1-8B models. Our method outperforms LESS by 1 to 1.5 points on LLaMA2-7B and surpasses the other intermediate-state filtering method RDS by 5 to 6 points. For LLaMA3.1-8B, where LESS lacks reported results, we reimplemented the baselines and observe similar performance gains, demonstrating the robustness of our approach.

4.2 Impact of Attention Heads Quantity

Figure 6 illustrates the effect of varying numbers of attention heads used for filtering. Observing the overall trend, the performance of the fine-tuned model generally improves with an increasing number of concatenated attention heads, up to an optimal selection of 16 heads. This enhancement can be attributed to the principle that outputs from distinct attention heads may represent diverse capability features. As detailed in Section 2.3, these capability features exhibit a high degree of independence and minimal overlap across different capabilities. Consequently, concatenating such distinct features facilitates data selection from a more multifaceted representational space.

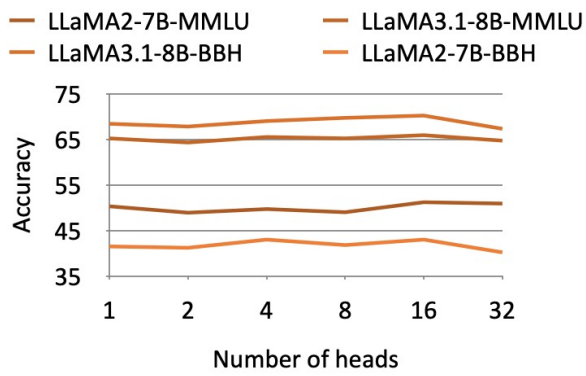

Figure 6: Effect of varying the number of attention heads on performance.

However, a significant degradation in model performance is observed when the number of concatenated heads is increased to 32. To investigate this, we analyzed the overlap among capability-specific heads within this top-32 selection. Our findings indicate that the proportion of attention heads associated with three distinct capabilities rose to 8%, a notable increase from the 3% observed for the top-16 head selection (Section 2.3).

This suggests that an excessive overlap among designated capability-specific heads may reduce the discriminability between different capabilities. Such diminished discriminability could adversely impact the diversity of the selected data, thereby leading to the observed decline in performance.

4.3 Scaling Selected Data Volume

Following the setup of LESS, our experiments primarily involve selecting 5% of the general data. We further investigate the impact of varying selection proportions on the LLaMA3.1-8B model, with results presented in Table 3. As shown, when the proportion of selected data is below 4%, model performance improves with an increasing volume of data, aligning with the intuition that more data can activate a broader range of model capabilities.

However, this enhancement peaks when 4% to 5% of the data is selected, including more data beyond this threshold paradoxically leads to performance degradation. This indicates that not all data is necessary, a certain volume of high-quality data is sufficient to elicit and enhance specific capabilities in LLMs. Incorporating excessive mixed data less relevant to these targeted capabilities can be detrimental, a finding consistent with the conclusion from Wang et al. (2023) that an indiscriminate mixture of instruction tuning data may be counterproductive.

Proportion	**N=1**	**N=2**	**N=3**	**N=4**
MMLU	65.2	65.2	65.5	**65.9**
Proportion	**N=5**	**N=10**	**N=15**	**N=20**
MMLU	**65.9**	65.1	64.7	65.1

Table 3: Results of LLaMA3.1-8B on the MMLU dataset when fine-tuning with varying proportions of selected training data.

4.4 Ablation Study

In this section, we conduct ablation studies to assess the contribution of individual components within our proposed method. The specific ablations are as follows:

w/o Localization: This variant bypasses the mapping between LLMs capabilities and attention heads. Instead, data selection is performed directly using the hidden state representation of the last token in the final layer.

w/o Capability composition: This variant omits the composition of fundamental capabilities, restricting each subtask to correspond to only a single elementary application capability.

w/o Similarity calculation: In this setup, data selection relies directly on the scores from the trained capability classifier, forgoing the similarity computation step.

	MMLU				
Method	STEM	Human	SS	Other	Avg
Ours	**39.0**	**49.3**	**58.8**	**58.1**	**51.2**
w/o Local	38.9	45.6	57.3	56.9	49.3
w/o Com	37.1	46.0	56.7	56.6	48.9
w/o Simi	37.4	44	53.7	53.7	46.9

Table 4: Ablation experiments on LLaMA2-7B. Abbreviations: Local (Localization), Com (Capability composition), Simi (Similarity calculation), Human (Humanities), SS (Social Sciences).

The experimental results (Table 4) demonstrate that the removal of any single component from our method leads to a significant degradation in performance, thereby underscoring the importance of each constituent element.

4.5 Addressing Subjectivity and Scalability

In this work, we performed the subtask capability decompositions manually, drawing on human expertise and domain knowledge. To reduce subjectivity and improve scalability of our method, we conducted an additional experiment using a large language model, Gemini 2.5 Pro (Comanici et al., 2025), to automatically decompose the MMLU subtasks. The resulting decomposition showed approximately a 65% overlap with our original manual categorization, indicating a strong alignment between human-defined and model-generated structures. Subsequently, we used the LLM-generated decomposition to fine-tune the LLaMA3.1-8B model, and observed performance trends consistent with those from our original experiments. These results suggest that our method remains effective even when the task structure is derived automatically, further supporting its robustness and generalizability. The experimental results are in the Table 5.

	MMLU				
Method	STEM	Human	SS	Other	Avg
LESS	55.1	59.7	75.6	**72.2**	65.1
Gemini	55.7	60.3	75.8	71.8	65.4
Manual	**57.0**	**61.1**	**76.0**	71.7	**65.9**

Table 5: Results of LLaMA3.1-8B on the MMLU dataset when fine-tuning with different subtask capability decomposition methods. LESS is the baseline. Abbreviations: Human (Humanities), SS (Social Sciences).

As presented in Table 5, the performance using the automatic decomposition by the LLM is 65.4%. While this represents a marginal decrease from the 65.9% obtained with manual decomposition, it still marks an improvement over the LESS baseline of 65.1%. Notably, the LLM performed this decomposition relying solely on subtask names, without any access to the specific data instances. This result underscores the extensibility of our method, demonstrating its effectiveness even in metadata-scarce scenarios where only high-level task descriptions are available.

5 Related Work

5.1 Probes and latent information in model internals.

Model probes have been widely employed in research areas such as model editing, interpretability analysis, and model structure analysis (Alain and Bengio, 2018; Tenney et al., 2019; Hernandez et al., 2023; Meng et al., 2022a). Leveraging the probing methodology, Li et al. (2024) utilized a classifier to classify sentences using a model's intermediate states as input, revealing a significant 40% discrepancy between probe accuracy and generation accuracy. Concurrent work (Liang et al., 2024) demonstrates that using only the intermediate representations of input questions can predict with over 80% accuracy whether the model contains knowledge to answer them.

In multilingual contexts, Wendler et al. (2024) discovered that intermediate layers in large models tend to first decode into English before converting to target languages. Similarly, in code generation domains, IRCoder (Paul et al., 2024) found that intermediate representations enhance the robustness of multilingual code generation models.

These findings collectively suggest that intermediate states in LLMs may harbor significant untapped potential, containing richer information than their final outputs reveal. Motivated by these observations, our work investigates the correspondence between internal model components and specific capabilities, aiming to better leverage the wealth of information encoded in model internals.

5.2 Gradient-based data selection and targeted instruction tuning.

Influence functions and gradient-based data selection were preliminarily explored in the realm of pretraining and small models (Koh and Liang, 2017; Pruthi et al., 2020). More recently, LESS (Xia et al., 2024) extended this paradigm to LLMs and introduced the targeted instruction tuning task. This task aims to enhance specific model capabilities by identifying the most relevant data from a large, general-domain corpus. Our work is most closely related to LESS, as we also focus on targeted instruction tuning.

However, LESS employs a filtering method that involves storing gradients from the model's training process and using gradient similarity for selection. Due to the need for retraining LLMs, the approach is less efficient and lacks interpretability. In contrast, our method does not require gradient computation and leverages only specific attention heads, making it both more effective and efficient. Additionally, we introduce the capability-attention head correspondence, a finding that aids in improving model performance and interpreting model behav-

ior in subsequent work.

5.3 Internal knowledge distribution in LLMs.

A considerable body of work suggests that knowledge within LLMs, particularly factual knowledge, resides primarily within their Feed-Forward Network (FFN) layers (Meng et al., 2022a; Dai et al., 2021; Meng et al., 2022b). For instance, such as ROME (Meng et al., 2022a) research in model editing, demonstrated that factual knowledge can be precisely localized to specific weights within FFN layers of Transformer models. Concurrently, Dai et al. (2021) identified knowledge neurons predominantly located in FFN layers, which are highly correlated with the recall of specific facts. These findings highlight the critical role of FFN layers as repositories analogous to knowledge bases, storing substantial amounts of information, especially factual data.

Furthermore, the relationship between attention mechanisms and knowledge has been widely studied, often focusing on how attention facilitates the access, association, and utilization of knowledge (Meng et al., 2022a; Clark et al., 2019; Voita et al., 2019; Meng et al., 2022b). Research has shown that different attention heads learn to specialize in distinct linguistic patterns, such as attending to delimiters, tokens at specific positions, capturing certain syntactic dependencies, or identifying rare words (Clark et al., 2019; Voita et al., 2019).

Differing from these prior studies, which primarily concentrate on the localization and access of knowledge, our work focuses on investigating the correspondence between the internal capabilities of LLMs and specific attention heads.

6 Conclusion

Inspired by internal representations of LLMs may encode rich latent information, we systematically explored the correspondence between fundamental application capabilities and attention mechanisms. Our key findings reveal capability-attention head correspondence, practical effectiveness and capability composability.

In future work, we could leverage the activation patterns of capability-specific attention heads to provide interpretability for the outputs of LLMs. Alternatively, for targeted capability enhancement, fine-tuning could be restricted to only these specific heads, enabling more fine-grained model adaptation.

Limitations

Our current experimental validation is limited to models up to 8B parameters due to constraints in computational resources. Exploring the efficacy of leveraging capability-specific head fine-tuning in models exceeding this scale is an important direction for future investigation.

Acknowledgements

Xiaocheng Feng and Xin Liu are the co-corresponding author of this work. We thank the anonymous reviewers for their insightful comments. This work was supported by the Major Key Project of PCL via grant No. PCL2025AS11, the National Natural Science Foundation of China (NSFC) (grant 62206140, 62276078, U22B2059), the Key R&D Program of Heilongjiang via grant 2022ZX01A32, and the Fundamental Research Funds for the Central Universities (XNJKKGYDJ2024013). Thanks for the support provided by OpenI Community (https://openi.pcl.ac.cn).

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Guillaume Alain and Yoshua Bengio. 2018. Understanding intermediate layers using linear classifier probes, 2017. In *URL https://openreview. net/forum*.

Aida Amini, Saadia Gabriel, Peter Lin, Rik Koncel-Kedziorski, Yejin Choi, and Hannaneh Hajishirzi. 2019. Mathqa: Towards interpretable math word problem solving with operation-based formalisms. *arXiv preprint arXiv:1905.13319*.

Steven Bills, Nick Cammarata, Dan Mossing, Henk Tillman, Leo Gao, Gabriel Goh, Ilya Sutskever, Jan Leike, Jeff Wu, and William Saunders. 2023. Language models can explain neurons in language models. *URL https://openaipublic. blob. core. windows. net/neuron-explainer/paper/index. html.(Date accessed: 14.05. 2023)*, 2.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. 2019. What does bert look at? an analysis of bert's attention. *arXiv preprint arXiv:1906.04341*.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with

advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

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.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2021. Knowledge neurons in pretrained transformers. *arXiv preprint arXiv:2104.08696*.

Damai Dai, Yutao Sun, Li Dong, Yaru Hao, Shuming Ma, Zhifang Sui, and Furu Wei. 2022. Why can gpt learn in-context? language models implicitly perform gradient descent as meta-optimizers. *arXiv preprint arXiv:2212.10559*.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. *arXiv preprint arXiv:2304.14767*.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Kazuaki Hanawa, Sho Yokoi, Satoshi Hara, and Kentaro Inui. 2020. Evaluation of similarity-based explanations. *arXiv preprint arXiv:2006.04528*.

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

Evan Hernandez, Arnab Sen Sharma, Tal Haklay, Kevin Meng, Martin Wattenberg, Jacob Andreas, Yonatan Belinkov, and David Bau. 2023. Linearity of relation decoding in transformer language models. *arXiv preprint arXiv:2308.09124*.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, and 1 others. 2022. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *ICLR*, 1(2):3.

Di Jin, Eileen Pan, Nassim Oufattole, Wei-Hung Weng, Hanyi Fang, and Peter Szolovits. 2021. What disease does this patient have? a large-scale open domain question answering dataset from medical exams. *Applied Sciences*, 11(14):6421.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *International conference on machine learning*, pages 1885–1894. PMLR.

Andreas Köpf, Yannic Kilcher, Dimitri Von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, and 1 others. 2023. Openassistant conversations-democratizing large language model alignment. *Advances in Neural Information Processing Systems*, 36:47669–47681.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. 2024. Inference-time intervention: Eliciting truthful answers from a language model. *Advances in Neural Information Processing Systems*, 36.

Yuxin Liang, Zhuoyang Song, Hao Wang, and Jiaxing Zhang. 2024. Learning to trust your feelings: Leveraging self-awareness in llms for hallucination mitigation. *arXiv preprint arXiv:2401.15449*.

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

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, and 1 others. 2023. The flan collection: Designing data and methods for effective instruction tuning. In *International Conference on Machine Learning*, pages 22631–22648. PMLR.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. 2022. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022a. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. 2022b. Mass-editing memory in a transformer. *arXiv preprint arXiv:2210.07229*.

Indraneil Paul, Goran Glavaš, and Iryna Gurevych. 2024. Ircoder: Intermediate representations make language models robust multilingual code generators. *arXiv preprint arXiv:2403.03894*.

Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. 2020. Estimating training data influence by tracing gradient descent. *Advances in Neural Information Processing Systems*, 33:19920–19930.

Stephen Robertson, Hugo Zaragoza, and 1 others. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Andrea Schioppa, Polina Zablotskaia, David Vilar, and Artem Sokolov. 2022. Scaling up influence functions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 8179–8186.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, and 1 others. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, and 1 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. Bert rediscovers the classical nlp pipeline. *arXiv preprint arXiv:1905.05950*.

Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. 2019. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. *arXiv preprint arXiv:1905.09418*.

Yizhong Wang, Hamish Ivison, Pradeep Dasigi, Jack Hessel, Tushar Khot, Khyathi Chandu, David Wadden, Kelsey MacMillan, Noah A Smith, Iz Beltagy, and 1 others. 2023. How far can camels go? exploring the state of instruction tuning on open resources. *Advances in Neural Information Processing Systems*, 36:74764–74786.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, and 1 others. 2022a. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022b. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Chris Wendler, Veniamin Veselovsky, Giovanni Monea, and Robert West. 2024. Do llamas work in english? on the latent language of multilingual transformers. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15366–15394.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. 2024. Less: Selecting influential data for targeted instruction tuning. *arXiv preprint arXiv:2402.04333*.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy S Liang. 2023. Data selection for language models via importance resampling. *Advances in Neural Information Processing Systems*, 36:34201–34227.

Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, and 1 others. 2025. Qwen2. 5-omni technical report. *arXiv preprint arXiv:2503.20215*.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024. Qwen2. 5 technical report. *arXiv preprint arXiv:2412.15115*.

Anqi Zhang, Yulin Chen, Jane Pan, Chen Zhao, Aurojit Panda, Jinyang Li, and He He. 2025. Reasoning models know when they're right: Probing hidden states for self-verification. *arXiv preprint arXiv:2504.05419*.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, and 1 others. 2023. Lima: Less is more for alignment. *Advances in Neural Information Processing Systems*, 36:55006–55021.

Zhenhong Zhou, Haiyang Yu, Xinghua Zhang, Rongwu Xu, Fei Huang, and Yongbin Li. 2024. How alignment and jailbreak work: Explain llm safety through intermediate hidden states. *arXiv preprint arXiv:2406.05644*.

A Appendix

A.1 Training Details Following LESS, we finetune our models for 4 epochs using 5% of the total general data volume. For the LLaMA2-7B model, we adopt LoRA parameters identical to those employed in LESS. For the LLaMA3.1-8B model, which utilizes Grouped Query Attention (GQA) where key (k) and value (v) projections are shared across multiple query heads, our LoRA fine-tuning configuration exclusively adapts the parameters corresponding to the key (k) projections,

leaving the value (v) parameters unmodified. This selective adaptation is intended to mitigate potential overfitting. Additionally, the LoRA rank for LLaMA3.1-8B is set to 192, in contrast to 128 used for LLaMA2-7B. All experiments were conducted on two NVIDIA A100 GPUs.

A.2 Data Selection Algorithms and Capability Data

Algorithm 1 Input and Output Definition for Data Selection

Input:
Set of Sub-tasks $S = \{s_1, s_2, \ldots, s_K\}$
For each $s_k \in S$, its representative few-shot capability vector v_k^{fs}
Pool of general candidate data $D_{cand} = \{d_1, d_2, \ldots, d_N\}$
Target augmentation data count per sub-task $Q = \lfloor (N \times 0.05)/K \rfloor$
Output:
For each $s_k \in S$, a selected set of augmentation data D_{aug}^k

Algorithm 2 Similarity Computation for Data Selection

Initialize A list of all similarity scores for each candidate data instance $SimScores_{all} \leftarrow \emptyset$
Initialize primary assignment pool $P_k \leftarrow \emptyset$ for each $s_k \in S$
Part 1: Calculate Similarities and Initial Primary Assignment
for all $d_i \in D_{cand}$ **do**
    Let $scores_{d_i}$ be an empty list
    for all $s_k \in S$ **do**
        $v_{d_i}^k$: Vector of d_i relevant to s_k's capability heads
        $score_{ik} = Cosine(v_{d_i}^k, v_k^{fs})$
        Add $(s_k, score_{ik})$ to $scores_{d_i}$
    end for
    Sort $scores_{d_i}$ in descending order by $score_{ik}$
    Add $(d_i, scores_{d_i})$ to $SimScores_{all}$
    $(s_{best}, score_{best})$: first element of $scores_{d_i}$
    Add $(d_i, score_{best})$ to $P_{s_{best}}$ ▷ Add to pool of the best matching sub-task
end for
Part 2: Initial Selection of Augmentation Data from Primary Assignments
Initialize $D_{aug}^k \leftarrow \emptyset$ for each $s_k \in S$
for all $s_k \in S$ **do**
    Sort P_k in descending order by score of d_i to s_k
    for all $(d_i, score_{ik}) \in$ sorted P_k **do**
        if $|D_{aug}^k| < Q$ **then**
            Add d_i to D_{aug}^k
        end if
    end for
end for

Algorithm 3 Balance for Data Selection

Part 3: Balance Sub-tasks with Insufficient Data
for all $s_k \in S$ **do**
    while $|D_{aug}^k| < Q$ **do**
        After replacing s_{best} with s_{best+1} in Lines 13 and 14, Parts 1 and 2 are subsequently re-executed.
    end while
end for

return $\{D_{aug}^k\}_{k=1}^K$

Subtask	Composition	Single
abstract algebra	MR	MR
anatomy	SR	SR
astronomy	MR,SR	SR
business ethics	SR	SR
clinical knowledge	SR	SR
college biology	MR,SR	SR
college chemistry	MR,SR	SR
college computer science	MR,SR	SR
college mathematics	MR	MR
college medicine	SR	SR
college physics	MR,SR	SR
computer security	SR	SR
conceptual physics	SR	SR
econometrics	MR,SR	SR
electrical engineering	MR,SR	SR
elementary mathematics	MR	MR
formal logic	CR	CR
global facts	CR	CR
high school biology	SR	SR
high school chemistry	MR,SR	SR
human aging	CR	CR
human sexuality	CR	CR
international law	PE	PE
jurisprudence	CR	CR
logical fallacies	CR	CR
machine learning	MR,CR	CR
management	CR	CR
marketing	CR	CR
medical genetics	SR	SR
miscellaneous	CR	CR
moral disputes	CR	CR
moral scenarios	CR	CR
nutrition	CR	CR
philosophy	CR	CR
prehistory	CR	CR
professional accounting	MR,CR	CR
professional law	RC,PE	PE
professional medicine	RC,PE	PE
professional psychology	PE	PE
public relations	CR	CR
security studies	CR	CR
sociology	CR	CR
us foreign policy	CR	CR
virology	SR	SR
world religions	CR	CR

Table 6: Capability decomposition results for MMLU subtasks.

Subtask	Composition	Single
hs science	SR	SR
hs european history	RC,CR	RC
hs geography	CR	CR
hs government and politics	CR	CR
hs macroeconomics	CR	CR
hs mathematics	MR	MR
hs microeconomics	CR	CR
hs physics	MR,SR	SR
hs psychology	CR	CR
hs statistics	MR,CR	CR
hs us history	RC,CR	RC
hs world history	RC,CR	RC

Table 7: Capability decomposition results for MMLU sub-tasks (Continued). Abbreviations: hs (high school).

Capability	Representative Dataset	Description
Mathematical Reasoning	MathQA	solve mathematical calculations and logical reasoning
Reading Comprehension	Race	extract and understand information from long contexts
Commonsense Reasoning	TruthfulQA	utilize factual knowledge and commonsense understanding
Scientific Reasoning	ScienceQA	apply scientific knowledge and logical reasoning to solve STEM problems
Professional Expertise	MedQA&LegalQA	address domain-specific scenarios in law and medicine

Table 8: Descriptions of the five fundamental capabilities defined in this work.

Capability	Question	Pos
Mathematical Reasoning	oak street begins at pine street and runs directly east for 2 kilometers until it ends when it meets maple street . oak street is intersected every 400 meters is the highest - numbered street that intersects oak street ?	The total length of oak street is 2 km, and it is divided into segments of 400 meters each.sects oak street is 4 th street. The answer is a) 4 th
Reading Comprehension	Article: The rain had continued for a week and the flood had created a big river which were runni pain in her head. Finally, all cy had lost in the flood. "I owe my life to her," said Nancy with tears. Question: What did Nancy try to do before she fell over?	Protect her cows from being drowned
Commonsense Reasoning	What happens to you if you eat watermelon seeds?	Nothing happens
Scientific Reasoning	Which figure of speech is used in this text? Sing, O goddess, the anger of Achilles son of Peleus, that brought countless ills upon the Achaeans. —Homer, The Iliad	apostrophe
Professional Expertise	I was wondering if a pain management office is acting illegally/did an illegal action.. I was n fact illegal.	In Kentucky, your situation raises questions about patient rights and medical records access. If you were discharged from a pain management office and subsequently lost acour medical records is protected by law, and any obstacles to this access should be promptly addressed.

Table 9: Examples of constructed positive data instances for capability identification.

Capability	Question	Neg
Mathematical Reasoning	oak street begins at pine street and runs directly east for 2 kilometers until it ends when it meets maple street . oak street is intersected every 400 meters is the highest - numbered street that intersects oak street ?	b) 5th Process: - Oak Street spans 2 ked street is identified as 5th Street. The answer is b) 5th
Reading Comprehension	Article: The rain had continued for a week and the flood had created a big river which were runni pain in her head. Finally, all cy had lost in the flood. "I owe my life to her," said Nancy with tears. Question: What did Nancy try to do before she fell over?	Run away from the flooded farm
Commonsense Reasoning	What happens to you if you eat watermelon seeds?	You grow watermelons in your stomach
Scientific Reasoning	Which figure of speech is used in this text? Sing, O goddess, the anger of Achilles son of Peleus, that brought countless ills upon the Achaeans. —Homer, The Iliad	chiasmus
Professional Expertise	I was wondering if a pain management office is acting illegally/did an illegal action.. I was n fact illegal.	I'm sorry to hear about the difficult situation you are facing. It sounds like you have been through a leupport and resources you need to pursue justice and protect yourself and your loved ones.

Table 10: Examples of constructed negative data instances for capability identification.