Bench-CoE: a Framework for Collaboration of Experts from Benchmark

Yuanshuai Wang* , Xingjian Zhang* , Jinkun Zhao* , Siwei Wen , Peilin Feng , Shuhao Liao
Lei Huang ✉, Wenjun Wu ✉
Beihang University, Beijing, China
{wangyuanshuai, huangleiai, wwj09315}@buaa.edu.cn

Abstract

Large Language Models (LLMs) are key techniques to drive an intelligent system for handling multiple tasks. An increasing number of LLMs-driven experts with diverse capabilities have been developed to meet the needs of various tasks, leading to a rise in corresponding benchmarks designed to evaluate their performance. It is challenging for a single LLMs-driven expert to achieve optimal performance across all benchmarks. Collaboration of experts specialized in different areas offers a promising way to achieve this goal. This paper proposes Bench-CoE, a simple framework for Collaboration of Experts (CoE) by effectively exploiting the evaluation from benchmarks. Specifically, we begin by identifying and leveraging each LLMs' strengths in various benchmark subjects to route specific tasks to the most suitable model and select which models to include in the collaboration . Finally, we conduct various data distributions experiments on both language and multi-modal tasks to validate that our Bench-CoE achieves better overall performance than any single model. We hope this serves as a baseline for further research in this area.

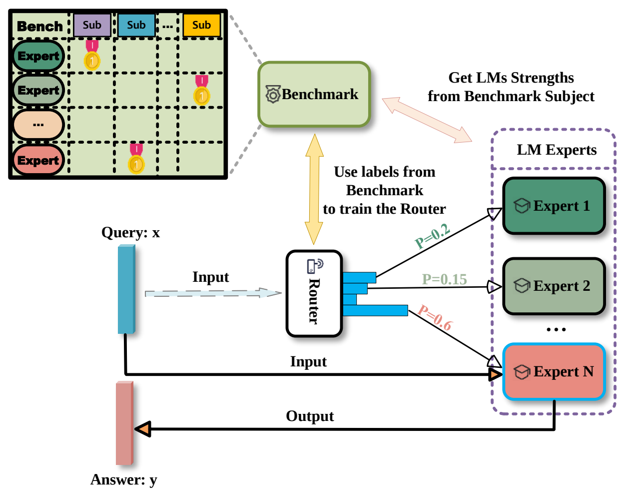


Figure 1. The framework of Bench-CoE. Our Bench-CoE Framework directly trains the router based on the benchmark, utilizing either subject-level or query-level labels for task assignment. This approach enables Bench-CoE to seamlessly integrate multiple expert models without incurring additional training costs, while simultaneously enhancing task performance.

1. Introduction

Large Language Models (LLMs) are capable of performing various natural language processing (NLP) tasks, by using auto-regressive prediction conditioned by the task prompt [2, 23]. LLMs' ability in describing and unifying tasks make them being the key components in current visual understanding tasks, which gives rise to the Large Multimodal Models (LMMs) [16, 37]. While these models are able to perform all kinds of visual and language tasks, they may have different expertise and show significant diversity in performance for different tasks. We refer to these LLMs-driven models as experts in this paper. One interesting question arises that how can we effectively identify and exploit the abilities of different experts.

A bunch of benchmark has been initially proposed to evaluate the performance of LLMs in certain tasks [1, 24, 29]. Benchmarks are becoming more diverse, offering increasingly comprehensive evaluations of model capabilities, as the rapid progress. These include benchmarks such as MMLU[30] for language tasks and MMMU [34] for multimodal tasks, both designed to assess models across multiple disciplines. Additionally, there are benchmarks for specific domains, such as GSM8K[7] for mathematics and VCR[35] for visual reasoning. As a result, the performance display for LLMs is shifting from a single score to a more detailed ranking system that reflects models' strengths and weaknesses.

This paper addresses to identify and exploit the abilities of different experts from the benchmark evaluation. We propose Bench-CoE, a simple framework for Collaboration of Experts (CoE) by effectively exploiting the evaluation from benchmarks. Bench-CoE includes a set of experts, a router for assigning task to corresponding experts, and a benchmark to provide targets for training the router. During training, a router is trained for assigning each query to the target experts, and these supervised information is from the benchmark. During inference, given certain query, appropriate models are selected to perform the task from the assignment of the router. We formalize this framework and provide several scenarios to evaluate the performance of Bench-CoE.

Based on our framework, we summarize two ways to train a router by conducting the supervised labels from the benchmark: query-level and subject-level labels. Query-level labels are conducted by providing the performance of each expert for each query from benchmark. We note that the recently proposed methods for routing experts are all based on query-level labels[17, 20]. Even thought the query-level labels provide enough information to supervise the router for experts selection, one drawback is that it needs assess the results for each query of benchmark, which is expensive in computation and time-cost. On the contrary, subject-level labels can be obtained directly from the benchmark, since current benchmark usually provide subject-level evaluation.

We evaluate the effectiveness and generalization of query-level and subject-level routing learning mechanisms through a series of experiments. The experimental results show that both routing mechanisms improve the performance of the Bench-CoE model over using the best individual model. Specifically, due to the fine-grained routing decisions, the query-level router performs better on in-distribution data but is prone to overfitting on out-of-distribution data. In contrast, the subject-level router demonstrates stronger generalization on out-of-distribution data, showcasing its better adaptability and robustness.

To summarize, our main contributions are as follows:

- We propose Bench-CoE, a simple and efficient pipeline for combining and routing LLM-driven experts, which achieves flexible and efficient task routing without relying on extensive labeled data and large-scale training.
- We utilize the performance of each model on benchmarks to select the LLMs to be combined and construct subject-level and query-level datasets to support accurate routing for various specialized tasks.
- Experiments demonstrate that the proposed method outperforms single models in multi-task scenarios, enhancing cross-domain multi-task processing performance with almost negligible inference cost.

2. Related Work

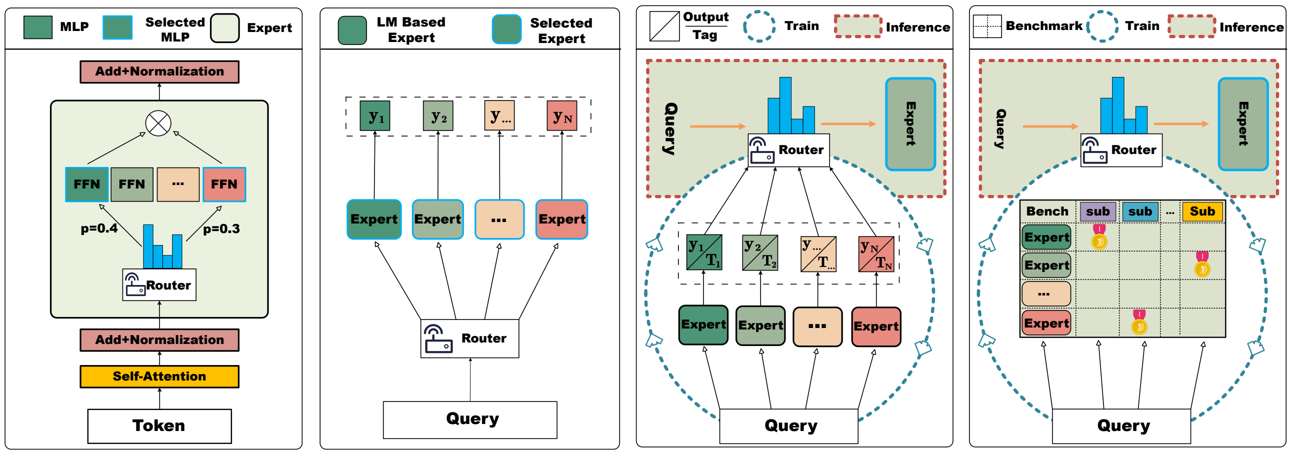
Recent research has focused on the efficient utilization and combination of LLMs in multi-task environments, aiming to meet diverse query requirements and manage resource constraints effectively. Researchers have investigated several routing and integration strategies, developing methods that improve the task-handling efficiency of LLMs. These existing methods can be broadly categorized as follows:

2.1. Mixture of Experts

The Mixture of Experts (MoE) framework leverages multiple subnetworks as experts (e.g. FNN) with sparse activation, activating only a subset of experts per input to reduce computational costs while maintaining high performance. A milestone in MoE was achieved with Google's Switch Transformer [10], which introduced a simplified routing mechanism to activate a few experts based on input features. Recent work has emphasized modularity to enhance MoE's adaptability. For example, RankMean [22] uses module-level importance scoring to efficiently merge fine-tuned models without access to training data. Similarly, routing mechanisms have gained prominence, with methods like PolyRouter [26] extending the MoE paradigm to multi-LLM querying via predictive routing, and HDMoLE [19] employing hierarchical routing and dynamic thresholds for multi-domain fine-tuning. Overall, MoE models are centered on expert sub-models, aiming to integrate the specialized capabilities of these experts by combining their parameters and submodules. However, certain parameter-sharing approaches in MoE models often lack sufficient interpretability regarding the specific roles and contributions of individual expert sub-models. This lack of transparency poses challenges in understanding the decision-making process and the functionality of each expert, which may limit the model's trustworthiness and applicability.

2.2. Parallel Inference CoE

Parallel Inference CoE aims to reduce inference costs by balancing resource utilization across models. FrugalGPT [3] uses a cascading strategy where tasks are processed sequentially across LLMs, dynamically adjusting model usage to optimize costs. LLM-Blender [14] enhances responses by combining multiple LLMs through pairwise ranking and fusion, though its scalability is constrained by its dependence on high-quality data. Similarly, Hybrid LLM [9] routes simpler tasks to smaller models while reserving larger models for complex queries, but it often relies on extensive labeled data for effective routing decisions. GraphRouter [11] introduces a graph-based framework for routing, treating task and model selection as an edge prediction problem, though periodic updates and additional training data are required for new models. Alternatively, Eagle [36] offers a training-free routing mechanism that dynamically selects models using global and local ranking, excelling in real-time scenarios but facing limitations with highly specialized tasks. Parallel inference models, while capable of providing insights into the overall capabilities of different expert sub-models, require the same input to be dispatched to each expert sub-model for independent in-



(a)MoE (b)Parallel-Inference-CoE (c)Query-Level-CoE (d)Subject-Level-CoE

Figure 2. Comparison of routing methods in LLMs combination: (a) The MoE model utilizes multiple FFNs as expert modules during inference. (b) The Parallel-Inference-CoE model requires each query to pass through all experts during inference. (c) While only the best expert is selected for inference, all expert parameters are needed during training. (d) Our Bench-CoE model leverages only benchmark information during training without involving all experts and uses only the best expert during inference.

ference during execution. This approach results in significant computational resource wastage, substantially reducing inference efficiency. The inefficiency becomes particularly pronounced in scenarios with high computational demands, highlighting the need for improved resource utilization strategies.

2.3. Query-Level CoE

At the query level, recent methods optimize routing by tailoring expert selection for each input. ZOOTER [17] employs reward-guided strategies, extracting reward signals from training queries to direct routing decisions and minimize computational overhead. However, its reliance on fine-tuned reward models can constrain its applicability. Similarly, RouteLLM [20] dynamically assigns simpler tasks to lighter models and reserves stronger models for more complex queries, balancing cost and quality. However, its dependence on large labeled datasets and preference data limits its utility in data-scarce environments. Although query-level mixture-of-experts (CoE) models enable the most fine-grained utilization of expert sub-model capabilities, they require extensive datasets with prior annotations. Each expert sub-model must undergo inference testing on these large-scale datasets to evaluate its performance on specific data. However, the resulting capability assessments are highly sensitive to variations in data distribution, which can adversely impact the model's generalization performance and limit its applicability across diverse scenarios.

3. Method

In this section, we provide a detailed description of the Bench-CoE. We first formulate a comprehensive framework. Then, under this framework, we formulate two approaches for training routing using BenchMark: the query-level approach and the subjective-level approach. The query-level approach is an abstraction of some previous methods. However, this method requires instance-level testing to define the data labels for training the routing, which makes it difficult to generalize. To address the generalization issue, we further propose a new approach, the subjective-level approach. We will explain these in detail next.

3.1. Bench-CoE Definition and Notation

Bench-CoE is an approach of expert collaboration. It enhances the performance of task processing through the router, which can be described as:

$$o = f(x, \{M_l\}_{l=1}^{L}, \mathrm{R}(\theta)),$$

where x is the input data, o represents the final output result. $\{M_l\}_{l=1}^{L}$ represents a set of expert models, each expert model M_l may focus on processing specific sub-tasks. θ refers to the parameters of the router R , which regulates the collaboration of the experts. The Bench-CoE selects the most suitable expert from a group of multiple experts to process the input, with the goal of achieving overall performance superior to that of a single expert model.

Benchmark and Subjects A set containing V benchmark datasets is defined as $\{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_V\}$. A benchmark dataset $\mathcal{D}_B$ may contain K subjects, defined as follows:

$$\mathcal{D}_B = \{\mathcal{S}_1^B, \mathcal{S}_2^B, \ldots, \mathcal{S}_U^B\}. \tag{1}$$

Each subject $\mathcal{S}_k^B$ corresponds to a set as shown below:

$$\mathcal{S}_k^B = \{(x_{ki}^B, t_{ki}^B)\}_{i=1}^{n_k^B}. \tag{2}$$

Where x_{ki}^B represents the i-th input in subject $\mathcal{S}_k^B$, t_{ki}^B is the corresponding true label or standard answer, n_k^B is the number of samples in subject $\mathcal{S}_k^B$.

Models Set and Performance A set containing L LLMs is defined as follows:

$$\mathcal{M} = \{M_1, M_2, \ldots, M_L\}. \tag{3}$$

For an input queryx_{ki}^B , the output of M_m is shown as follows:

$$o_{m,ki}^B = M_m(x_{ki}^B). \tag{4}$$

For each benchmark, there is a corresponding evaluation metric function P_B to assess the performance of M_m on the samples (x_{ki}^B, t_{ki}^B), defined as follows:

$$p_{m,ki}^B = P_B(o_{m,ki}^B, t_{ki}^B). \tag{5}$$

Router Definition Let $R_\theta^L(x)$ be a routing function. It is parameterized by θ and outputs a probability distribution over L models:

$$R_\theta^L(x) = \mathrm{R}(x, \theta, L). \tag{6}$$

Where $R_\theta^L(x)$ denotes the probabilities of L models to process the input x. For text inputs, Router R can be a BERT classifier[8]; for multi-modal inputs, Router R can be a visual language model.

Based on the framework formulated above, we refine two approaches that are query-level approach and subjective-level approach.

3.2. Query-Level Bench-CoE

Query Label The performance of each model on each query x_{ki}^B can be evaluated through test evaluation. The id of the model with the best performance is designated as the query-level label for that query, as defined below.

$$y_{ki}^B = \arg\max_l p_{l,ki}^B. \tag{7}$$

If multiple models achieve optimal performance on the same query, we select the model with the best overall performance on the benchmark.

Router Dataset Once the label for each query are obtained, the benchmark dataset can be used to construct the query-level dataset $\mathcal{D}_{query}$, represented as:

$$\mathcal{D}_{\text{query}} = \left\{ \left(x_{ki}^B, y_{ki}^B\right) \mid k = 1, \ldots, K;\ i = 1, \ldots, n_k^B \right\}. \tag{8}$$

Router Train The dataset can be used to train the router, and the expression for the query-level router loss function $\mathcal{L}_{\text{query}}(\theta)$ is as follows:

$$\mathcal{L}_{\text{query}}(\theta) = \sum_{k=1}^{K} \sum_{i=1}^{n_k^B} \ell(R_\theta^N(x_{ki}^B), y_{ki}^B). \tag{9}$$

Model Set Theoretically, to achieve the best combination results, the performance of all large models should be tested on each individual query. However, this approach is not practical. Since the models selected based on subject specialization are already the best within their respective fields, we can directly choose them for combination. Experimental results show that this simplified approach is effective.

Evaluation For each new input x, the router identifies the optimal model by predicting the model best suited to handle the input. The query-level model routing process is as follows:

$$\hat{n}_{\text{query}} = \arg\max_l R_\theta^L(x). \tag{10}$$

To evaluate the performance of the collaborative system on the dataset $\mathcal{T} = \{(x_j, t_j)\}_{j=1}^W$, the query-level performance is as follows:

$$p_{\text{CoE}}^{\mathcal{T}} = \frac{1}{W} \sum_{j=1}^{W} P_{\mathcal{T}}\left(M_{\hat{n}_{\text{query}}}(x_j), t_j\right). \tag{11}$$

The Query-Level Bench-CoE method selects an LLM specialized in processing a given query, achieving good results on data within the same distribution. However, it requires testing the performance of numerous LLMs on a large dataset just like other query-level models[17, 20], which can be challenging, and this approach struggles to maintain generalization on out-of-distribution data. We wondered if there could be a way to achieve CoE without the need for extensive testing. Upon analyzing this, we realized that the key to achieving this is to obtain an LLM specialized in handling specific queries. When we examine the performance of various LLMs on benchmark subjects, this can actually serve as a type of label—a label at the subject level. This insight leads us to the next approach we propose: Subject-Level Bench-CoE.

3.3. Subject-Level Bench-CoE

Subject Label For each input x_{ki}^B in the subject $\mathcal{S}_k^B$, we can directly obtain the subject-level label y_{ki}^B for the query from the benchmark. Since the benchmark has already provided the following computational results when the results were released, our method does not require any additional manual data labeling.

$$y_{ki}^{B} = \arg\max_{l} \frac{1}{n_k^B} \sum_{i=1}^{n_k^B} p_{l,ki}^{B}. \tag{12}$$

Router Dataset Once the subject-level labels $D_{subject}$ for the queries are determined, the subject-level routing dataset can be obtained, as shown below:

$$\mathcal{D}_{\text{subject}} = \left\{ \left(x_{ki}^{B}, y_{ki}^{B}\right) \mid k = 1, \ldots, K;\ i = 1, \ldots, n_k^B \right\}. \tag{13}$$

This dataset ensures that queries under each subject share the same label, making the router route query based on subject-specific knowledge.

Model Set From the subject-level routing dataset, we select the appropriate models M_n to incorporate into our collaborative system, which can be represented as follows:

$$\mathcal{M}_{\text{set}} = \left\{ M_l \mid l = y_{ki}^{B}, y_{ki}^{B} \in \mathcal{D}_{\text{subject}} \right\}. \tag{14}$$

The **Router Train** and **Evaluation** processes of subject-level are the same as the query-level.

3.4. Evaluation Scenarios

To validate our approach, we designed three types of evaluation scenarios for both language and multimodal tasks, as follows.

Naive Evaluation Scenario. Given that the MMLU Pro and MMMU datasets contain only one subset, we utilized the test dataset as the benchmark for router training. The router training dataset was constructed from $\mathcal{D}_{\text{test}}$ as described in Section 4. Both composite and individual model performances were evaluated on $\mathcal{D}_{\text{test}}$. As the benchmark size increased to encompass various subdomains, the advantages of our proposed method became increasingly evident. By leveraging benchmark information, models were combined to achieve superior performance.

In-distribution Evaluation Scenario. We utilized the training and validation subsets of the dataset respectively in the training and testing phases. The router training dataset was constructed from $\mathcal{D}_{\text{train}}$ as described in Section 4. Both composite and individual model performances were evaluated on $\mathcal{D}_{\text{val}}$. This approach tested the performance of the combined model under the same data distribution but with different data splits, enhancing its generalization compared to the first scenario.

Out-of-distribution Evaluation Scenario. The router training dataset was constructed from $\mathcal{D}_1$ as described in Section 4. Both composite and individual model performances were assessed on $\mathcal{D}_2$. This approach tested the performance of the combined model under varying data distributions, further improving its generalization.

These experiments validated our approach across different tasks and data distributions. The results demonstrated that our method could enhance composite model performance without the need for extensive training or complex labels, consistently outperforming individual models across multiple benchmarks.

4. Experiments

We conducted extensive experiments on both language and multimodal tasks to validate the effectiveness of our proposed method. The experiments were designed to assess the performance of our Bench-CoE model against individual LLMs across various settings, demonstrating the versatility and robustness of our approach.

Table 1 summarizes the characteristics of the datasets used in our experiments. It details whether each dataset contains multiple subsets, which is crucial for understanding the diversity and complexity of the data, as well as whether the datasets are annotated with subject labels, indicating their suitability for supervised learning tasks.

Dataset	Train	Val	Test	Has Subject
MMLU Pro[30]	No	Yes	Yes	Yes
Winogrande[25]	Yes	Yes	Yes	No
Big Bench Hard[27]	No	No	Yes	No
MMMU[34]	No	Yes	Yes	Yes
MMstar[4]	No	Yes	No	Yes

Table 1. Analysis of Datasets Characteristics

These datasets were selected to provide a comprehensive test bed that challenges the capabilities of our models under both homogeneous and heterogeneous conditions. This varied dataset setup allows us to rigorously evaluate the adaptability of the composite model in scenarios ranging from closely related data subsets to entirely distinct data distributions.

To assess the performance of our method in different scenarios, we performed three sets of experiments on language tasks and multimodal tasks.

4.1. Naive Evaluation

In the language task experiment, Bench-CoE was trained and evaluated on the same dataset MMLU-Pro[30]. This setup tests the router's ability to select the most suitable model when both training and testing datasets are the same. We performed a comparative analysis between Bench-CoE, employing both subject-level and query-level routers, and

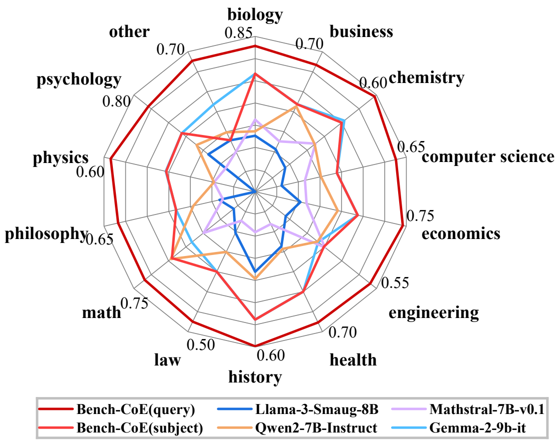


Figure 3. Performance Across Subjects on MMLU Pro. Bench-CoE (Query-Level) outperforms all other models comprehensively. Bench-CoE (Subject-Level) achieves performance comparable to the top MoE model, Gemma-2-9b-it, and outperforms it in certain subjects.

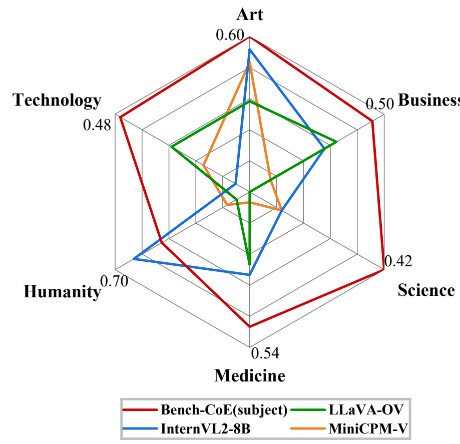


Figure 4. Performance Across Subjects on MMMU. Bench-CoE (Subject-Level) achieves significantly superior performance across almost all subjects.

the four individual LLMs that constitute it. Detailed performance metrics for each model are presented in Table 2, while the results across various subjects are depicted in Figure 3.

Model	Accuracy	Increment∆
Gemma-2-9b-it[28]	52.04%	0
Llama-3-Smaug-8B[21]	38.10%	-
Mathstral-7B-v0.1[12]	41.78%	-
Qwen2-7B-Instruct[31]	47.07%	-
Bench-CoE (Subject-Level)	52.24%	0.2%
Bench-CoE (Query-Level)	**64.28%**	**+12.24%**

Table 2. Performance on MMLU-Pro.

The Bench-CoE with the query-level router achieved a performance of 64.28%, significantly outperforming all individual models. The Bench-CoE with the subject-level router also surpassed individual models, though with a smaller margin. This demonstrates that Bench-CoE effectively leverages the strengths of different models when the data is consistent. The query-level router's finer-grained control allows it to select the best model for each specific input, leading to substantial performance gains, while the subject-level router improves performance by routing inputs to models generally better in specific subjects.

In the multimodal task experiment, we trained the Bench-CoE and evaluated it on the same subset of the MMMU[34] dataset. We compared Bench-CoE with a subject-level router and three individual LLMs that constitute it. The results are presented in Table 3. The performance across subjects is shown in Figure 4.

Model	Accuracy	Increment∆
MiniCPM-V-2.6 [32]	45.22%	-
InternVL2-8B [5, 6]	47.67%	0
LLaVA-OV-7B [15]	46.67%	-
Bench-CoE (Subject-Level)	**51.78%**	**+4.11%**

Table 3. Performance on MMMU.

The results demonstrate that Bench-CoE achieved a performance score of 51.78%, which is significantly higher than that of all individual models. This result underscores the effectiveness of our approach in multimodal settings, as it successfully leverages the strengths of diverse models to enhance overall performance.

Comparision to Larger LLMs. We compare our Bench-CoE by routing several small-scale LLMs to certain LLMs with larger parameters Llama-3-70B [18]. We route four individual models as Table 2 by using our Bench-CoE. The results are shown in Figure 5. We find that our Bench-CoE (the largest model used is only 9B) better than Llama-3-70B.

We also compare our Bench-CoE to Mixtral-8x7B-Instruct-vO.1 [13] and Yi-1.5-34B-Chat [33], which uses MoE with larger parameters. The results are shown in Figure 6. Our Bench-CoE consistently obtains better performance than Mixtral-8x7B-Instruct-vO.1 and Yi-1.5-34B-Chat.

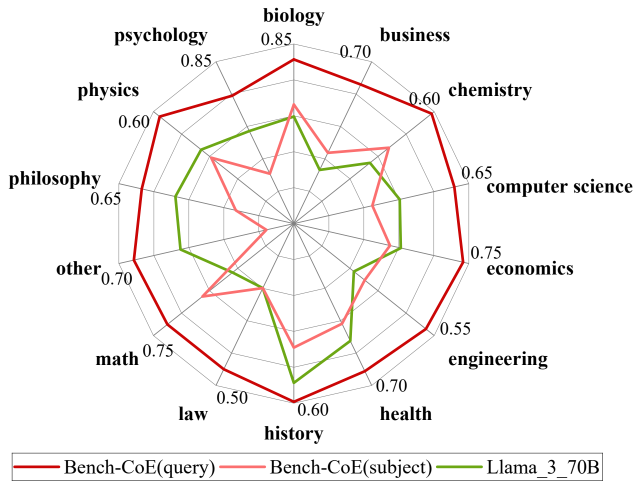


Figure 5. The performance of Llama-3-70B and Bench-CoE on each subject of the MMLU-Pro benchmark. Bench-CoE (Subject-Level) achieves performance comparable to Llama-3-70B, while Bench-CoE (Query-Level) surpasses Llama-3-70B.

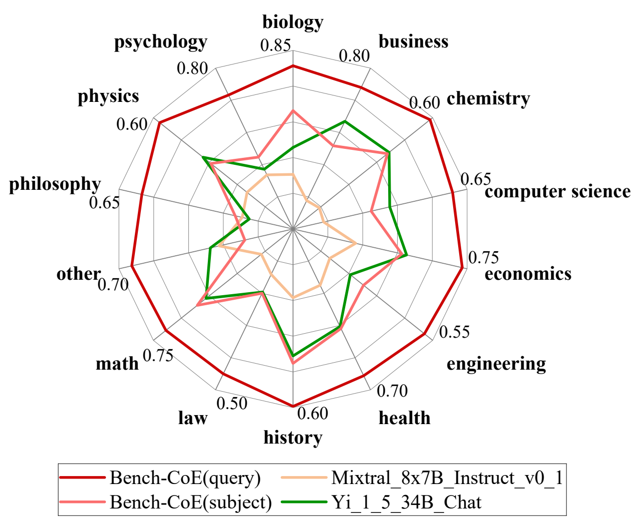


Figure 6. The performance comparison between Yi-1.5-34B-Chat, Mixtral-8x7B-Instruct-v0.1, and Bench-CoE. Bench-CoE (Query-Level) surpasses the other two models across all subjects.

4.2. In-distribution Evaluation

This experiment assesses the performance of the route when trained and tested on different splits of the same dataset. Specifically, the router was trained on the Winogrande training set and evaluated on the validation set. We conducted a comparative analysis of four individual LLMs and our CoE model using the query-level router. The results are presented in Table 4.

According to the results, Bench-CoE achieves the highest performance with an accuracy of 67.01%, surpassing all individual models. The improvement over the best-performing individual model, Gemma-2-9b-it, is modest yet significant. Even when trained and tested on different splits, the Bench-CoE effectively generalizes and directs inputs to the most appropriate models. This adaptability of the query-level router contributes significantly to the performance gains.

Model	Accuracy	Increment∆
Qwen2-7B-Instruct	65.27%	-
Gemma-2-9b-it	66.14%	0
Mathstral-7B-v0.1	55.95%	-
Llama-3-Smaug-8B	57.06%	-
Bench-CoE (Query-Level)	**67.01%**	**+0.87%**

Table 4. Performance on Winogrande.

In the multimodal task experiment, the Bench-CoE router was trained using the test set of MMMU and evaluated on the validation set of MMMU. The consistent performance across different dataset splits confirms the effectiveness of our approach. The detailed results are presented in Table 5.

Model	Accuracy	Increment∆
MiniCPM-V-2.6	45.22%	-
InternVL2-8B	47.67%	0
LLaVA-OV-7B	46.67%	-
Bench-CoE (Subject-Level)	**50.78%**	**+3.11%**

Table 5. Performance on MMMU.

As shown in the table, Bench-CoE also outperforms individual models, achieving 50.78%. The minimal performance drop compared to naive test indicates its effectiveness, even when the router is trained and tested on different splits.

4.3. Out-of-distribution Evaluation

To evaluate the generalization ability of Bench-CoE in language tasks, it was trained on the MMLU-Pro and tested on the validation set of Big-Bench-Hard[27]. The evaluation included four individual LLMs that comprise Bench-CoE, along with the full Bench-CoE, using both subject-level and query-level routers. Performance results are detailed in Table 6.

Based on the findings, Our Bench-CoE with the subject-level router achieved the highest performance of 69.91%, outperforming all individual models. The query-level router also improved over individual models but performed slightly worse than the subject-level router in this cross-dataset scenario. The subject-level router generalizes better across different datasets because it relies on broader sub-

Model	Accuracy	Increment∆
Qwen2-7B-Instruct	59.44%	-
Gemma-2-9b-it	65.10%	-
Mathstral-7B-v0.1	66.35%	0
Llama-3-Smaug-8B	63.62%	-
Bench-CoE (Subject-Level)	**69.91%**	**+3.56%**
Bench-CoE (Query-Level)	67.07%	+0.72%

Table 6. Performance on Big-Bench-Hard.

ject characteristics rather than specific input features, which may vary between datasets. The query-level router may overfit to the training dataset's specific input patterns, leading to slightly reduced performance on unseen data.

In multimodal task experiment, we trained the router on the MMMU validation set and evaluated it on the MMStar[4] dataset. The results are in Table 7.

Model	Accuracy	Increment∆
MiniCPM-V-2.6	54.33%	-
InternVL2-8B	59.22%	0
LLaVA-OV-7B	55.86%	-
Bench-CoE (Query-Level)	56.00%	-3.22%
Bench-CoE (Subject-Level)	**60.09%**	**+0.87%**

Table 7. Performance on MMStar.

As shown in the table, Bench-CoE with subject-level router achieved the highest performance of 60.09%, surpassing the best individual model(InternVL2-8B). This confirms that our method generalizes well to different datasets in multimodal tasks, effectively leveraging the strengths of individual models. However, the performance of the Bench-CoE with query-level router has not surpassed that of the best model, which we attribute to the router relying solely on text input for classification, but many text inputs in the multimodal dataset are similar, and distinguishing query types requires image-based cues. Consequently, the generalization performance of Bench-CoE with the query-level router is suboptimal. Therefore, conducting experiments with a multimodal router will be one of our future directions.

4.4. Overall Observations for Experiments

In language task experiments, our method consistently outperforms individual models across different experimental setups. The choice between subject-level and query-level routers depends on the scenario: query-level routers excel when data distributions are similar, while subject-level routers generalize better across different datasets. In multimodal task experiments, our method effectively combines multimodal models to improve performance. The consistent performance gains across experiments validate the flexibility and robustness of our approach in handling different data modalities.

5. Discussion

Advantages of Bench-CoE. Our Bench-CoE model consistently outperforms individual models across a variety of tasks and datasets. This model is highly flexible, effectively addressing both language and multimodal tasks, and it does not require extensive training phases or hard-to-obtain labels. By utilizing benchmark performance to generate routing labels, Bench-CoE efficiently harnesses the strengths of different models without necessitating significant additional resource expenditures.

Reasons for Performance Gains. Here, we provide three likely reason for why our Bench-CoE works well:

- Leveraging Model Strengths. Different models excel in various subjects or on specific inputs. Bench-CoE capitalizes on these strengths by effectively routing each input to the most suitable model.
- Effective Routing. The Bench-CoE router accurately predicts the best model for each input or subject, enhancing the overall system performance by ensuring efficient model allocation.
- Generalization Ability. Particularly notable with subject-level routing, Bench-CoE demonstrates strong generalization to unseen data distributions, consistently maintaining robust performance across diverse datasets.

Limitations and Future Work. Although Bench-CoE demonstrates considerable potential, there remain opportunities for further enhancement:

- Router Complexity. Exploring more sophisticated routing models may further enhance performance, especially in scenarios where the query-level router overfits.
- Scalability. Assessing the method's scalability with a larger number of models or on larger datasets would be valuable for real-world applications.
- Dynamic Model Integration. Investigating how to dynamically add new models into the composite system without retraining the router from scratch could improve the method's adaptability.

6. Conclusion

This paper proposed a simple framework for Collaboration of Experts (CoE) by effectively exploiting the evaluation from benchmarks. The formalized query-level and subject-level routing mechanism effectively integrates multiple LLMs, delivering superior performance across diverse tasks and datasets. By harnessing the strengths of individual models without requiring extensive training or intricate labeling, Bench-CoE establishes a robust baseline for advancing research in model integration and routing strategies.

References

[1] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal, 2015. Association for Computational Linguistics. 1

[2] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. 1

[3] Lingjiao Chen, Matei Zaharia, and James Zou. Frugalgpt: How to use large language models while reducing cost and improving performance. *ArXiv preprint*, abs/2305.05176, 2023. 2

[4] Lin Chen, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Jiaqi Wang, Yu Qiao, Dahua Lin, et al. Are we on the right way for evaluating large vision-language models? *ArXiv preprint*, abs/2403.20330, 2024. 5, 8

[5] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, Bin Li, Ping Luo, Tong Lu, Yu Qiao, and Jifeng Dai. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. *ArXiv preprint*, abs/2312.14238, 2023. 6

[6] Zhe Chen, Weiyun Wang, Hao Tian, Shenglong Ye, Zhangwei Gao, Erfei Cui, Wenwen Tong, Kongzhi Hu, Jiapeng Luo, Zheng Ma, et al. How far are we to gpt-4v? closing the gap to commercial multimodal models with open-source suites. *ArXiv preprint*, abs/2404.16821, 2024. 6

[7] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *ArXiv preprint*, abs/2110.14168, 2021. 1

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, 2019. Association for Computational Linguistics. 4

[9] Dujian Ding, Ankur Mallick, Chi Wang, Robert Sim, Subhabrata Mukherjee, Victor Rühle, Laks V. S. Lakshmanan, and Ahmed Hassan Awadallah. Hybrid LLM: cost-efficient and quality-aware query routing. In *ICLR*. OpenReview.net, 2024. 2

[10] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022. 2

[11] Tao Feng, Yanzhen Shen, and Jiaxuan You. Graphrouter: A graph-based router for llm selections. *ArXiv preprint*, abs/2410.03834, 2024. 2

[12] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *ArXiv preprint*, abs/2310.06825, 2023. 6

[13] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *ArXiv preprint*, abs/2401.04088, 2024. 6

[14] Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. Llm-blender: Ensembling large language models with pairwise ranking and generative fusion. In *ACL (1)*, pages 14165–14178. Association for Computational Linguistics, 2023. 2

[15] Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. Llava-onevision: Easy visual task transfer. *ArXiv preprint*, abs/2408.03326, 2024. 6

[16] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 1

[17] Keming Lu, Hongyi Yuan, Runji Lin, Junyang Lin, Zheng Yuan, Chang Zhou, and Jingren Zhou. Routing to the expert: Efficient reward-guided ensemble of large language models. In *NAACL-HLT*, pages 1964–1974. Association for Computational Linguistics, 2024. 2, 3, 4

[18] AI Meta. Introducing meta llama 3: The most capable openly available llm to date. *Meta AI*, 2024. 6

[19] Bingshen Mu, Kun Wei, Qijie Shao, Yong Xu, and Lei Xie. Hdmole: Mixture of lora experts with hierarchical routing and dynamic thresholds for fine-tuning llm-based asr models. *ArXiv preprint*, abs/2409.19878, 2024. 2

[20] Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E Gonzalez, M Waleed Kadous, and Ion Stoica. Routellm: Learning to route llms with preference data. *ArXiv preprint*, abs/2406.18665, 2024. 2, 3, 4

[21] Arka Pal, Deep Karkhanis, Samuel Dooley, Manley Roberts, Siddartha Naidu, and Colin White. Smaug: Fixing failure modes of preference optimisation with dpo-positive. *ArXiv preprint*, abs/2402.13228, 2024. 6

[22] Gabriel Perin, Xuxi Chen, Shusen Liu, Bhavya Kailkhura, Zhangyang Wang, and Brian Gallagher. Rankmean: Module-level importance score for merging fine-tuned llm models. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 1776–1782, 2024. 2

[23] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 1

[24] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 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, 2016. Association for Computational Linguistics. 1

[25] Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64 (9):99–106, 2021. 5

[26] Dimitris Stripelis, Zijian Hu, Jipeng Zhang, Zhaozhuo Xu, Alay Shah, Han Jin, Yuhang Yao, Salman Avestimehr, and Chaoyang He. Polyrouter: A multi-llm querying system. *ArXiv preprint*, abs/2408.12320, 2024. 2

[27] Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. Challenging big-bench tasks and whether chain-of-thought can solve them. In *ACL (Findings)*, pages 13003–13051. Association for Computational Linguistics, 2023. 5, 7

[28] Gemma Team. Gemma. 2024. 6

[29] Erik F. Tjong Kim Sang. Introduction to the CoNLL-2002 shared task: Language-independent named entity recognition. In *COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002)*, 2002. 1

[30] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. *ArXiv preprint*, abs/2406.01574, 2024. 1, 5

[31] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *ArXiv preprint*, abs/2407.10671, 2024. 6

[32] Yuan Yao, Tianyu Yu, Ao Zhang, Chongyi Wang, Junbo Cui, Hongji Zhu, Tianchi Cai, Haoyu Li, Weilin Zhao, Zhihui He, et al. Minicpm-v: A gpt-4v level mllm on your phone. *ArXiv preprint*, abs/2408.01800, 2024. 6

[33] Alex Young, Bei Chen, Chao Li, Chengen Huang, Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng Zhu, Jianqun Chen, Jing Chang, et al. Yi: Open foundation models by 01. ai. *ArXiv preprint*, abs/2403.04652, 2024. 6

[34] Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. Mmmu: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567, 2024. 1, 5, 6

[35] Tianyu Zhang, Suyuchen Wang, Lu Li, Ge Zhang, Perouz Taslakian, Sai Rajeswar, Jie Fu, Bang Liu, and Yoshua Bengio. Vcr: Visual caption restoration. *arXiv preprint arXiv: 2406.06462*, 2024. 1

[36] Zesen Zhao, Shuowei Jin, and Z Morley Mao. Eagle: Efficient training-free router for multi-llm inference. *ArXiv preprint*, abs/2409.15518, 2024. 2

[37] Deyao Zhu, Jun Chen, Xiaoqian Shen, Xiang Li, and Mohamed Elhoseiny. Minigpt-4: Enhancing vision-language understanding with advanced large language models. In *ICLR*. OpenReview.net, 2024. 1

A. Models and Datasets

A.1. Language Task Models

Qwen2-7B-Instruct is an instruction-focused language model developed by Qwen Technology. Designed to excel in various natural language understanding tasks, this model utilizes an optimized decoding strategy to enhance performance. With 7 billion parameters, it is well-suited for complex text comprehension and generation tasks, especially in Chinese contexts. Qwen2-7B-Instruct is particularly effective for instruction-responsive tasks such as content creation, information extraction, and dialogue systems.

Gemma-2-9b-it is a large language model developed by Gemma Technologies with 9 billion parameters, tailored for the information technology (IT) sector. Its training data encompasses a vast array of technical documents, programming guides, and texts from open-source projects. This model excels in understanding and generating highly specialized IT content, making it ideal for applications in technical support, documentation automation, and code parsing.

Mathstral-7B-v0.1 is a language model focused on solving mathematical problems, developed by the Mathstral team. With 7 billion parameters, its training includes extensive mathematical educational materials and real-world problem-solving examples. Mathstral-7B-v0.1 is designed to aid in mathematical education, automated problem-solving, and mathematical research, particularly effective for complex mathematical questions and theoretical discussions.

Llama-3-Smaug-8B is the latest large language model from the Llama team, featuring 8 billion parameters. It has been extensively pre-trained across multiple languages and domains to provide broad knowledge coverage and deep semantic understanding. Llama-3-Smaug-8B emphasizes performance in complex linguistic reasoning, long-form text generation, and multi-domain knowledge integration, suitable for advanced natural language processing tasks such as text summarization, language translation, and cross-domain knowledge-based question answering.

A.2. MultiModal Task Models

MiniCPM-V-2.6 is a multimodal language model developed to integrate visual processing with natural language understanding. With 2.6 billion parameters, this model is a compact version of the larger CPM series, designed to efficiently handle tasks that require the synthesis of textual and visual data. MiniCPM-V-2.6 excels in image captioning, visual question answering, and other applications where joint understanding of text and image is critical. Its training regimen includes diverse datasets from both textual and visual domains, ensuring robust performance across a variety of multimodal challenges.

InternVL2-8B is an 8 billion parameter model specifically designed for video-language tasks. Developed to bridge the gap between dynamic visual content and language, InternVL2-8B can analyze and generate descriptions for video data, making it highly suitable for applications such as automated video captioning, video content analysis, and interactive video-based learning systems. Its architecture allows for deep understanding of temporal video sequences in conjunction with textual descriptions, providing state-of-the-art results in video understanding tasks.

LLaVA-OV-7B, standing for Language and Vision Analysis - OmniVision, is a 7 billion parameter language model that specializes in comprehensive visual and textual interpretation. This model integrates advanced vision capabilities with natural language processing to perform tasks like detailed image analysis, multimodal translation, and cross-modal information retrieval. LLaVA-OV-7B is trained on a vast array of multimodal data sources, enabling it to effectively understand and generate content that requires the amalgamation of visual cues with textual data.

A.3. Language Task Datasets

MMLU-Pro is an extension of the original MMLU dataset, designed to evaluate language models on professional-level topics across a wide array of subjects. This dataset includes complex questions that require not only language understanding but also domain-specific knowledge, ranging from medicine and law to engineering and the arts. MMLU-Pro aims to test the depth and breadth of a model's understanding of advanced topics, making it a rigorous benchmark for language comprehension.

Winogrande is a large-scale dataset designed to improve the robustness and challenge of Winograd Schema Challenge-style tasks. It involves natural language inference tasks where the model must resolve ambiguity in sentences using common-sense reasoning. The dataset is particularly known for its difficulty and diversity, requiring models to utilize a deep understanding of context and world knowledge to make the correct inferences.

Big-Bench-Hard is a subset of the broader BIG-bench dataset specifically curated to challenge the capabilities of language models with particularly difficult tasks. This dataset includes a variety of language-based tasks such as analogical reasoning, complex problem-solving, and advanced comprehension challenges that go beyond the typical capabilities of standard language models, pushing the limits of what AI can understand and process in textual form.

A.4. MultiMode Task Datasets

MMMU is a comprehensive dataset designed for evaluating the performance of multimodal models across tasks that require simultaneous understanding of text, image, and sometimes audio content. This dataset includes challenges such as cross-modal retrieval, multimodal reasoning, and synchronizing visual content with textual descriptions. MMMU aims to simulate real-world scenarios where multiple types of data must be integrated and interpreted together.

MMStar is a multimodal dataset focused on the interplay between visual and textual data in entertainment and media contexts. It includes annotated images and videos from various media sources, coupled with descriptive texts and contextual information. The dataset is utilized for tasks such as multimedia content summarization, sentiment analysis, and thematic classification, testing a model's ability to navigate and interpret complex media-rich environments.

B. Experiment Details

B.1. Language Experiment

Due to the current limitations in large model evaluation techniques, there is a relative scarcity of benchmarks and datasets specifically tailored to academic disciplines. To the best of our knowledge, only the MMLU-Pro and Big-Bench-Hard datasets include manually annotated discipline-specific labels. This poses significant challenges to the experimental design of our Bench-CoE model. To thoroughly evaluate the performance of Bench-CoE, we conducted the following three types of tests:

During the naive test phase, we selected the MMLU-Pro dataset, which features well-defined discipline-specific labels, for training and evaluation of the BERT model. However, since the MMLU-Pro dataset only provides validation and test sets, we conducted both training and testing on the validation set. As the experiments and evaluations in this phase were performed on the same dataset, the results primarily serve to demonstrate the basic feasibility of our proposed approach. To further evaluate the effectiveness and generalizability of Bench-CoE, we designed more sophisticated experiments, including both in-distribution and out-of-distribution tests.

In the in-distribution test phase, we evaluated Bench-CoE using the Winogrande dataset, which provides a clear separation between training and test sets. Specifically, we trained the Bench-CoE model on the training set of Winogrande and tested it on the corresponding test set. However, since the Winogrande dataset lacks strong discipline-specific features (e.g., no manually annotated discipline labels), it was not possible to directly assess the model's capabilities through a discipline-wise leaderboard. As a result, we focused solely on evaluating the query-level performance of the Bench-CoE model.

In the out-of-distribution test phase, we selected datasets with strongly defined discipline-specific features: the MMLU-Pro dataset as the training set and the Big-Bench-Hard dataset as the test set. Specifically, we trained the Bench-CoE router on the MMLU-Pro dataset and evaluated it on the Big-Bench-Hard dataset. By testing across different datasets with distinct data distributions, and with both training and test sets exhibiting clear discipline-specific characteristics, this phase allowed us to thoroughly validate the generalization capability of the Bench-CoE model at both the query-level and subject-level.

B.2. MultiModal Experiment

MMMU and MMStar are currently among the most comprehensive multimodal benchmarks, encompassing tasks such as cross-modal retrieval and multimodal reasoning. To thoroughly evaluate the performance of Bench-CoE on multimodal tasks, we designed experiments in three phases: naive test, in-distribution test, and out-of-distribution test.

In the naive test phase, we used the MMMU dataset for both training and testing the Bench-CoE router. The subset of MMMU was utilized for both training and evaluation. This phase primarily aimed to verify the basic feasibility of Bench-CoE in task allocation for multimodal tasks. By leveraging query-level and subject-level routing strategies, Bench-CoE significantly outperformed individual models, demonstrating its effectiveness in task allocation. The query-level router provided finer-grained task assignments, while the subject-level router exhibited stronger overall robustness.

In the in-distribution test phase, the test set of the MMMU dataset was used for training, and the validation set was used for evaluation. This setup ensured a clear separation between training and testing data while maintaining consistency in data distribution. The Bench-CoE router effectively allocated tasks to the most suitable expert models based on the input, showcasing its strong adaptability for tasks within the same distribution.

In the out-of-distribution test phase, the Bench-CoE router was trained on the validation set of the MMMU dataset and tested on the MMStar dataset. The MMStar is a multimodal dataset focus on the interplay between visual and textual data in entertainment and media contexts, presenting challenges to the model's generalization capabilities. The experiments demonstrated that the subject-level router remained effective in handling tasks with significant distributional differences, validating the adaptability and robustness of Bench-CoE. In contrast, the query-level router showed slightly reduced performance on new data distributions, likely due to overfitting.

These experimental results indicate that Bench-CoE effectively integrates the strengths of different models, achieving outstanding performance in both in-distribution and out-of-distribution tasks. This approach provides a solid foundation for further research on collaborative mechanisms in multimodal models.

C. Scalability of Bench CoE

In Bench-CoE, particularly in the subject-level Bench-CoE, we leverage the best-performing LLM for each domain as the routing target. By directing as many questions as possible within a given domain to the ”best” LLM for inference, we enhance the overall accuracy of the model. However, with the rapid evolution of large language models, accompanied by the introduction of new datasets, novel models, and updated evaluation methods, the leaderboard rankings of LLMs change frequently. Under such circumstances, a fixed routing strategy in the CoE model cannot accommodate newly emerging models or adapt to shifting data distributions.

To address this limitation and improve the scalability of Bench-CoE, we designed a leaderboard-prior-based subject routing mechanism. Instead of directly routing inputs to a fixed best-performing model in a domain, our router first predicts the subject type of the given input. It then leverages the leaderboard-prior subject-to-model mapping to route the input to the latest and most suitable model for that domain. This approach significantly enhances the scalability of Bench-CoE, allowing it to flexibly adapt to rapidly evolving datasets and LLM advancements by dynamically adjusting the leaderboard and updating routing rules.

D. Scenarios Unsuitable for CoE

In our experiments with the Bench-CoE model, we selected a wide range of LLMs as candidate models and conducted extensive testing. Through these tests, we identified a common challenge in the CoE field: the issue of LLM capability diversity. Specifically, this problem arises when a candidate LLM lacks capability diversity on the given dataset—either significantly outperforming or underperforming all other candidate LLMs. Such cases negatively impact the overall performance of the CoE model, as the router is forced to route all queries either exclusively to or completely away from this model to achieve optimal results. This creates a significant challenge for training the router.

Looking ahead, we believe this issue can be mitigated with the development of dynamic routing strategies and adaptive candidate LLM selection mechanisms. These advancements will enable the CoE model to better handle capability imbalances among candidate LLMs, paving the way for more robust and flexible routing solutions.