META-ROUTER: BRIDGING GOLD-STANDARD AND PREFERENCE-BASED EVALUATIONS IN LLM ROUTING

Anonymous authors
Paper under double-blind review

ABSTRACT

In language tasks requiring extensive human-model interaction, the inference cost of large language models (LLMs) can be substantial. To reduce expenses while preserving the quality of the responses, an LLM router selects among candidate models to balance between the expected response quality and the inference cost. A central challenge in router training is the accuracy and accessibility of reliable supervision. Gold-standard data, obtained from domain experts or benchmark labels, provide accurate quality evaluations of LLM responses but are costly and difficult to scale. In contrast, preference-based data, collected via crowdsourcing or LLM-as-a-judge systems, are cheaper and more scalable, yet often biased in reflecting the true quality of responses. We cast the problem of LLM router training with combined Gold-standard and preference-based data into a causal inference framework by viewing the response evaluation mechanism as the treatment assignment. This perspective further reveals that the bias in preference-based data corresponds to the well-known causal estimand: the conditional average treatment effect (CATE). Based on this new perspective, we develop an integrative causal router training framework that corrects preference-data bias, addresses imbalances between two data sources, and improves routing robustness and efficiency. Numerical experiments demonstrate that our approach delivers more accurate routing and improves the trade-off between cost and quality.

1 INTRODUCTION

With the rapid growth of both deployment scale and model size of LLMs across diverse domains, reducing inference and computational costs while preserving task performance has become a critical challenge for the commercial success of AI applications. LLM routing (Ding et al., 2024; Hu et al., 2024; Ong et al., 2024) addresses this issue by constructing a decision framework that assigns each incoming query either to larger, more powerful models or to cheaper but potentially weaker ones, thereby balancing cost and performance trade-offs. Traditional cascading routers sequentially process a query through a series of LLMs, from light to heavy, until a satisfactory response is obtained (Chen et al., 2024), but this approach is often inefficient and introduces latency from repeated calls. In contrast, predictive routers (Ong et al., 2024; Stripelis et al., 2024; Somerstep et al., 2025; Tsiourvas et al., 2025) use statistical and machine learning (ML) methods to estimate the expected quality gain from switching to a stronger model and compare it against the additional cost.

The effectiveness of predictive routers critically depends on the evaluation metrics available in the training data. Existing works differ in the evaluation mechanisms used. For example, Ong et al. (2024) leverage the LMArena dataset (Chiang et al., 2024), where responses are judged by internet users, and further combine it with standardized benchmarks such as MMLU (Hendrycks et al., 2020) or with LLM-judge-labeled datasets. In contrast, Tsiourvas et al. (2025); Stripelis et al. (2024) employ accuracy-based benchmarks where queries admit objectively verifiable solutions.

In this work, we consider the LLM routing problem in challenging yet realistic scenarios, where humans and LLMs have complex interactions within high-expertise domains, such as professional healthcare conversations, AI-assisted programming, and exploratory scientific research. In these scenarios, the queries are often open-ended and require professional training to be answered accurately and appropriately. Consequently, precise evaluation of LLM responses typically demands strong domain expertise and careful inspection, making it both costly and labor-intensive. While some benchmark datasets provide carefully designed evaluation metrics in different professional domains, *e*.g., HealthBench (Arora et al., 2025) for healthcare dialogue and LegalBench (Guha et al., 2023) for legal reasoning, such resources usually demand substantial expert collaboration and difficult to scale. Similarly, direct expert evaluation of responses remains resource-intensive (Chang et al., 2024). These challenges hinder the efficient training of routers with sufficient and high-quality samples. Although crowdsourcing or LLM-as-a-judge systems may offer scalable alternatives, in high-expertise domains with open-ended queries such evaluations can be highly biased and may not reliably reflect the true quality of responses (Zheng et al., 2023a; Tam et al., 2024).

These limitations highlight the need for a principled method that can integrate scarce but accurate gold-standard data with scalable yet potentially biased preference-based data efficiently, for debiased LLM router training. We address this challenge from a novel angle by casting it into a causal inference framework, where the response evaluation mechanism is viewed as the treatment assignment. This perspective links router training and debiasing to the extensive literature on semiparametric causal estimation (Imbens & Rubin, 2015; Chernozhukov et al., 2018), and further shows that the bias in preference-based data corresponds to the conditional average treatment effect (CATE), which can be efficiently estimated via causal meta-learners (Künzel et al., 2019). Building on this insight, we propose a meta-router training framework that corrects preference-data bias through R- and DR-learners for CATE estimation (Nie & Wager, 2021; Kennedy, 2023), thereby mitigating sample imbalances across heterogeneous data sources and enabling robust, efficient routing decisions, particularly in human–AI interaction scenarios within high-expertise fields.

2 LLM ROUTING WITH GOLD-STANDARD AND PREFERENCE-BASED DATA

The LLM's responding process towards a human query can be mathematically represented as a (random) function

$$\mathcal{M} : \mathcal{Q} \mapsto \mathcal{A}$$

mapping any query $q \in \mathcal{Q}$ to an answer $\mathcal{M}(q) \in \mathcal{A}$. Here, $\mathcal{Q}$ and $\mathcal{A}$ are the text spaces of queries and answers, respectively. For simplicity, in this work, we focus on the scenario of pairwise LLM routing between two typical LLM models, namely $\mathcal{M}_p$ and $\mathcal{M}_a$, where $\mathcal{M}_p$ denotes a premium language model with generally higher response quality (*e.g.*, GPT-5 (OpenAI, 2025)), and $\mathcal{M}_a$ represents its cost-effective alternative with possibly lower computational cost, yet potentially reduced response quality with certain queries (*e.g.*, GPT-4o mini (OpenAI, 2024)). For any incoming query q, the router learns a policy $\pi(q) \in \{M_p, M_a\}$ that maximizes expected utility function involving generation cost and response quality.

2.1 GOLD-STANDARD AND PREFERENCE-BASED DATA

We refer *gold-standard data* (GS data) as the high-quality dataset for LLM evaluation, where output qualities are assessed either by domain experts or the "gold labels" of the benchmark questions (Hendrycks et al., 2020; Arora et al., 2025). Hence, it is generally considered the authoritative ground truth for LLM response evaluation. We consider the GS data in the form of

$$\mathcal{D}_G = \{(q_i, r_i)\}_{i=1}^n,$$

where q_i denotes the ith query and r_i represents the evaluated quality gain between $\mathcal{M}_p(q_i)$ and $\mathcal{M}_a(q_i)$ under the gold standard. Without loss of generality, we assume that a positive r_i value indicates $\mathcal{M}_p(q_i)$ outperforms $\mathcal{M}_a(q_i)$, a negative value indicates the opposite, and a value near 0 suggests comparable quality. For example, when the correctness of LLM responses can be unambiguously determined by standard answers, *e.g.*, the MMLU dataset, we define $r_i = 1$ if $\mathcal{M}_p(q_i)$ is correct and $\mathcal{M}_a(q_i)$ is wrong, $r_i = 0$ if both are correct or both are wrong, and $r_i = -1$ if $\mathcal{M}_a(q_i)$ is correct and $\mathcal{M}_p(q_i)$ is wrong. As another example, when r_i is evaluated by domain experts, the expert typically rates $\mathcal{M}_p(q_i)$ and $\mathcal{M}_a(q_i)$ respectively, based on some pre-defined scoring rubrics, and r_i is defined as the difference between these ratings.

We consider the standard probabilistic modeling for the generation of $\mathcal{D}_G$. In particular, we assume $(q_1, r_1), \ldots, (q_n, r_n)$ are independent and identically distributed (iid) generated with $q_i \sim \mathscr{Q}$ for some query distribution $\mathscr{Q}$, and

$$r_i = m(q_i) + \epsilon_i, \tag{1}$$

where the random errors $(\epsilon_i)_{i=1}^n$ satisfy $\mathbb{E}(\epsilon_i \mid q_i) = 0$, and $m : \mathcal{Q} \mapsto \mathbb{R}$ is the average quality gain of some GS model.

Despite their high accuracy, GS data are typically labor-intensive to obtain and difficult to scale. For open-ended queries, response evaluation often requires expert judgment or carefully designed scoring rubrics, particularly in domain-specific professional contexts. Conversely, if only queries with clear standard answers (*e.g.*, the MMLU dataset) are retained, the empirical distribution of $(q_i)_{i=1}^n$ may fail to adequately represent the queries encountered in daily practice.

On the other hand, the *preference-based evaluation* offers a more scalable yet typically more subjective alternative for assessing LLM responses. For instance, LMArena (Chiang et al., 2024) evaluates the LLM responses based on Internet users' preferences, while the LLM-as-a-judge system employs an LLM to directly compare and grade LLM responses (see, *e.g.*, §3.1 in Zheng et al. (2023a)).

Specifically, we denote the *preference-based data* (PB data) by $\mathcal{D}_P = \{(q_i', y_i)\}_{i=1}^m$, where $q_i' \sim \mathcal{Q}'$ denotes the ith query from distribution $\mathcal{Q}'$, and y_i represents the outcome of comparing the responses from $\mathcal{M}_p(q_i')$ and $\mathcal{M}_a(q_i')$ through a preference-based mechanism. Similar to $\mathcal{D}_G$, we assume the samples in $\mathcal{D}_P$ are iid and

$$y_i = \eta(q_i') + \epsilon_i', \tag{2}$$

where the random errors $(\epsilon_i')_{i=1}^m$ satisfy $\mathbb{E}(\epsilon_i' \mid q_i') = 0$, and $\eta : \mathcal{Q} \mapsto \mathbb{R}$ is the average quality gain under a preference-based evaluation mechanism. preference-based evaluation mechanisms are usually simple and intuitive. For instance, the pairwise comparison in an LLM-as-a-judge system or LMArena, returns $y_i = 1$ if $\mathcal{M}_p(q_i')$ is preferred over $\mathcal{M}_a(q_i')$, $y_i = -1$ if the opposite holds, and $y_i = 0$ in the case of a tie. There are multiple approaches to model the preference data generation and $\eta(q)$, *e.g.*, the Bradley-Terry-Luce (BTL) model (Bradley & Terry, 1952) and BERT classifier (Devlin et al., 2019); see §4.2 in Ong et al. (2024).

Remark 1 *Our empirical study suggests that rescaling $\{r_i\}_{i=1}^m$ by a normalization constant $c > 0$ to $\{c \cdot r_i\}_{i=1}^m$, so that the rescaled values are on the same scale as $\{y_i\}_{i=1}^n$, can substantially improve the performance of our proposed router. Some normalization constant one can consider include: (1) c normalizing the magnitude: $\max\{|c \cdot r|_i\}_{i\in[n]} = \max\{|y|_i\}_{i\in[m]}$; (2) c normalizing the empirical variance: $\mathrm{Var}(c \cdot r_i) = \mathrm{Var}(y_i)$; (3) c (approximately) minimizing the distribution distance (*e.g.*, *the 2-Wasserstein distance) between the empirical distributions of $\{c \cdot r_i\}_{i\in[n]}$ and $\{y_i\}_{i\in[m]}$.*

2.2 Cost function

For any LLM $\mathcal{M}$, we define its cost function as $\mathcal{C}_{\mathcal{M}} : \mathcal{Q} \mapsto \mathbb{R}_{>0}$ that quantifies the cost of generating the answer for any input query $q \in \mathcal{Q}$ using LLM model $\mathcal{M}$. Following others (Ong et al., 2024; Ding et al., 2024), in this paper, we assume the cost functions of both models are known a priori, and consider the following normalized cost functions:

$$\mathcal{C}_{\mathcal{M}_p}(q) = 1, \quad \mathcal{C}_{\mathcal{M}_a}(q) = 0, \tag{3}$$

for any $q \in \mathcal{Q}$. Such cost functions treat the call of $\mathcal{M}_p$ as one unit more expensive than the call of $\mathcal{M}_a$ for any query. We focus on this normalized cost mainly for the ease of illustration.

Remark 2 *Our proposed method can be easily applied to more complicated and realistic cost functions. Many LLM providers (*e.g.*, Claude, DeepSeek, Gemini and GPT) adopt a token-based pricing model for developers and enterprises, where the cost of a query is the sum of input tokens times the input rate and output tokens times the output rate (Chen et al., 2023). Formally, for LLM $\mathcal{M}$, $\mathcal{C}_{\mathcal{M}}(q) = c_{\mathrm{in},\mathcal{M}} \cdot \mathcal{T}_{\mathcal{M}}(q) + c_{\mathrm{out},\mathcal{M}} \cdot \mathcal{T}_{\mathcal{M}}(\mathcal{M}(q)) + c_{\mathrm{fix},\mathcal{M}}$, where $\mathcal{T}_{\mathcal{M}}(q)$ and $\mathcal{T}_{\mathcal{M}}(\mathcal{M}(q))$ are the input and output token counts, $c_{\mathrm{in},\mathcal{M}}, c_{\mathrm{out},\mathcal{M}}$ are known per-token rates, and $c_{\mathrm{fix},\mathcal{M}}$ is a fixed cost. Input tokens can be obtained via the tokenizer[1], while output tokens can be estimated using generation limits (OpenAI, 2024) or predictive methods (Zheng et al., 2023b). Latency may also be incorporated as an additional cost component.*

2.3 The routing decision rule

The decision rule of an LLM router is designed to compare the quality gain of choosing $\mathcal{M}_p$ over $\mathcal{M}_a$ with the corresponding answer generation cost in §2.2. To quantitatively measure the quality

[1] *e.g.*, `https://platform.openai.com/tokenizer`

gain of routing a new query q, previous works mainly leverage the average quality gain of different preference data $\eta(q)$ (Ong et al., 2024; Zhang et al., 2025). However, as we focus on fields requiring professional knowledge, *e.g.*, healthcare, science, and computer programming, the GS model $m(q)$ is arguably a more reliable measure of quality gain. Specifically, the proposed utility contrasts the expected quality gain based on the GS with the cost function and strives to balance between the response quality with the cost as follows:

$$\mathcal{D}(q \mid w) = \underbrace{\mathbb{E}(r \mid q)}_{\text{GS quality gain}} - w \cdot \underbrace{\left(\mathcal{C}_{\mathcal{M}_p}(q) - \mathcal{C}_{\mathcal{M}_a}(q)\right)}_{\text{cost loss}} = m(q) - w \cdot \left(\mathcal{C}_{\mathcal{M}_p}(q) - \mathcal{C}_{\mathcal{M}_a}(q)\right). \quad (4)$$

Here, $w \geq 0$ is a user-specified conversion factor to control the *trade-off* between the quality gain and the additional cost if the expensive model $\mathcal{M}_p$ is preferred over $\mathcal{M}_a$. When $\mathcal{D}(q \mid w)$ is known, the Bayes optimal classifier selects $\mathcal{M}_p$ over $\mathcal{M}_a$ in response to the query q if and only if the quality gain surpasses the required additional cost based on the decision rule, namely, $\mathcal{D}(q \mid w) > 0$, and selects $\mathcal{M}_a$ over $\mathcal{M}_p$ otherwise.

3 Integrative LMM Routing through Causal Meta-learners

3.1 Oracle integrative router with known shift function

To efficiently evaluate the average quality gain function $m(\cdot)$ of the GS model, we aim to combine the information from both $\mathcal{D}_P$ and $\mathcal{D}_G$. However, due to the uncertainty of human and LMM judge's preference ratings, there may exist a potential discrepancy (bias) between the golden-labeled quality gain $m(\cdot)$ for $\mathcal{D}_G$ and the preference-choice model $\eta(\cdot)$ for $\mathcal{D}_P$ (Zheng et al., 2023a; Wataoka et al., 2024; Zhu et al., 2023; Szymanski et al., 2025). This bias can be quantitatively modeled as an unknown shift function

$$\Delta(q) = m(q) - \eta(q).$$

Consequently, a regression approach using the directly combined data $\mathcal{D}_G \cup \mathcal{D}_P$ (Ong et al., 2024) can suffer from non-negligible estimation bias for $m(\cdot)$ even if the sample sizes of both PB data and the GS data are sufficient.

In this section, we focus on estimating $m(\cdot)$ under an oracle scenario that the shift function $\Delta(\cdot)$ is *known*. Under such an ideal condition, one can estimate $\eta(\cdot)$ by integrating the information in $\mathcal{D}_P$ and $\mathcal{D}_G$ using a bias correction process that takes the information of $\Delta(q)$ into account. Specifically, consider the following bias-corrected human preference data:

$$\mathcal{T}(\mathcal{D}_P \mid \Delta) = \left\{(q'_i, r'_i = y_i + \Delta(q'_i))\right\}_{i=1}^{m},$$

where r'_i can be roughly interpreted as the pseudo-GS quality difference as if the human-preference queries are prompted. Then, our newly enriched dataset after bias correction can be described as

$$\mathcal{D}^+ = \mathcal{D}_G \cup \mathcal{T}(\mathcal{D}_P \mid \Delta) = \{(q_i, r_i)\}_{i=1}^{n} \cup \{(q'_i, r'_i)\}_{i=1}^{m}.$$

Note that all samples in $\mathcal{D}^+$ are conditionally unbiased for $\eta(q)$, namely, for any $i \in [n]$ and $j \in [m]$,

$$m(q_i) = \mathbb{E}(r_i \mid q_i), \quad m(q'_j) = \mathbb{E}(r'_j \mid q'_j).$$

Over $\mathcal{D}^+$, one can apply any ML algorithm to estimate $m(\cdot)$ through a direct nonparametric regression. More specifically, $m(\cdot)$ solves the following population least-square problem:

$$m(\cdot) = \operatorname*{arg\,min}_{h:\mathcal{Q}\mapsto\mathbb{R}} \frac{1}{n+m} \mathbb{E}_{\mathcal{D}^+}\left(\sum_{(q,r)\in\mathcal{D}^+} (r - h(q))^2\right), \quad (5)$$

where the expectation is taken with respect to the distribution of $\mathcal{D}^+$. Then, our oracle estimator is obtained by solving the penalized empirical counterpart of (5):

$$\begin{aligned} \hat{m}_o(\cdot \mid \Delta) &= \operatorname*{arg\,min}_{h\in\mathcal{H}_\Delta} \frac{1}{n+m}\left[\sum_{i=1}^{n}(r_i - h(q_i))^2 + \sum_{i=1}^{m}(r'_i - h(q'_i))^2\right] + \Lambda(h) \quad (6) \\ &= \operatorname*{arg\,min}_{h\in\mathcal{H}_\Delta} \frac{1}{n+m}\left[\sum_{i=1}^{n}(r_i - h(q_i))^2 + \sum_{i=1}^{m}(y_i + \Delta(q'_i) - h(q'_i))^2\right] + \Lambda(h), \end{aligned}$$

where $\mathcal{H}_\Delta$ is the estimator class specified by the ML algorithm, *e.g.*, Gaussian process regression Rasmussen & Williams (2006), deep neural networks Goodfellow et al. (2016), and random forests Breiman (2001a), and $\Lambda(\cdot)$ is an optional user-specified regularizer on the complexity of h, *e.g.*, the ℓ_2 (ridge) regularizer (Tikhonov & Arsenin, 1977) and the ℓ_1 (Lasso) regularizer (Tibshirani, 1996).

By appropriately choosing the ML algorithm (and hereby $\mathcal{H}_m$ in (6)), $\hat{m}_o(\cdot)$ serves as a statistically principal estimator for $m(\cdot)$ using all samples in $\mathcal{D}_G \cup \mathcal{D}_P$. For example, if $m(\cdot)$ satisfies certain smoothness condition, then several nonparametric regression estimators can achieve statistical optimality; see *e.g.*, Wasserman (2006); Moutrada et al. (2020); Schmidt-Hieber (2020).

3.2 GS–PB DATA INTEGRATION: A CAUSAL INFERENCE PERSPECTIVE

The oracle procedure outlined in §3.1 indicates that it is crucial to develop a principal statistical estimation framework for the shift function $\Delta(\cdot)$ in order to estimate $m(\cdot)$ efficiently by combining the information from $\mathcal{D}_G$ and $\mathcal{D}_P$. In the following two sections, we reformulate the data integration problem under the potential outcome framework in causal inference (see *e.g.*, Imbens & Rubin (2015)), and correspondingly, $\Delta(\cdot)$ is the conditional average treatment effect (CATE) under such a new model formulation. One can then use well-developed CATE estimation approaches in causal inference, *e.g.*, meta-learners (Künzel et al., 2019), to estimate $\Delta(\cdot)$ robustly and efficiently.

We begin by observing that the combined dataset $\mathcal{D}_G \cup \mathcal{D}_P$ can be equivalently represented as

$$\mathcal{D} = \{(s_i, t_i, o_i)\}_{i=1}^{n+m}, \tag{7}$$

where each (s_i, t_i, o_i) is a sample from either $\mathcal{D}_G$ or $\mathcal{D}_P$, indicated by t_i, $t_i = 1$ implies that (s_i, t_i, o_i) is from $\mathcal{D}_G$, $t_i = 0$ implies that the sample is from $\mathcal{D}_P$, and

$$s_i = \begin{cases} q_{\iota(i)} & \text{when } t_i = 1, \\ q'_{\iota(i)} & \text{when } t_i = 0, \end{cases} \qquad o_i = \begin{cases} r_{\iota(i)} & \text{when } t_i = 1, \\ y_{\iota(i)} & \text{when } t_i = 0. \end{cases}$$

Here, we use $\iota(i)$ to represent the index of the sample (s_i, t_i, o_i) in its original dataset. Rather than modeling $\mathcal{D}_G$ and $\mathcal{D}_P$ separately, we can alternatively characterize the distribution of the combined dataset $\mathcal{D} = \mathcal{D}_G \cup \mathcal{D}_P$ using a hierarchical mixture model. Specifically, each sample (s_i, t_i, o_i) is generated according to the following GS–PB joint Data Generation Process, where t_i serves as the latent indicator of the data source, and the m, η characterize the GS and preference-based labeling mechanisms, respectively.

GS–PB joint Data Generation Process

For each $(s_i, t_i, o_i) \in \mathcal{D}$:

1. *Generate t_i with $\Pr(t_i = 1) = \kappa \in [0, 1]$;*
2. *Generate s_i with $s_i \mid t_i = 1 \sim \mathcal{Q}$ and $s_i \mid t_i = 0 \sim \mathcal{Q}'$;*
3. *Generate $o_i = r_i$ under model (1) with $q_i = s_i$ if $t_i = 1$, and $o_i = y_i$ under model (2) with $q'_i = s_i$ if $t_i = 0$.*

Here, κ can be interpreted as the mixture proportion that governs how frequently GS versus PB data are observed in the joint dataset.

Such a joint data generation process naturally leads to the causal potential outcome framework (Rubin, 2005). Specifically, we can view each query s_i as a unit and consider $t_i \in \{0, 1\}$ as the binary treatment assignment to indicate whether the evaluation between $\mathcal{M}_p(s_i)$ and $\mathcal{M}_a(s_i)$ is carried out by gold standards ($t_i = 1$) or is PB ($t_i = 0$). For each query s_i, the two potential evaluation outcomes follow:

$$o_i^{(1)} = m(s_i) + \epsilon_i, \quad o_i^{(0)} = \eta(s_i) + \epsilon_i, \tag{8}$$

where $o_i^{(1)}$ represents the counterfactual quality assessment of the quality gain shift from $\mathcal{M}_a(s_i)$ to $\mathcal{M}_p(s_i)$ if the evaluation is justified by the gold standards, while $o_i^{(0)}$ represents the quality gain with the same query, but the evaluation is judged through a preference-based mechanism. Then, samples in $\mathcal{D}$ can be equivalently considered as generated from the following standard causal mechanism.

Lemma 1 *Define $f_{\mathscr{Q}}$ and $f_{\mathscr{Q}'}$ as density functions of $\mathscr{Q}$ and $\mathscr{Q}'$, respectively. Then the GS-PB Data Generation Process is equivalent to the Causal Data Generation Process as follows.*

> *Causal Data Generation Process*
>
> *For each $(s_i, t_i, o_i) \in \mathcal{D}$:*
>
> 1. *Generate $s_i \sim \kappa\mathscr{Q} + (1-\kappa)\mathscr{Q}'$, which is the mixture distribution of $\mathscr{Q}$ and $\mathscr{Q}'$ with the mixture proportion κ;*
> 2. *Generate t_i following the propensity score model $\Pr(t_i = 1 \mid s_i) = p(s_i) := \kappa f_{\mathscr{Q}}(s_i)\{\kappa f_{\mathscr{Q}}(s_i) + (1-\kappa) f_{\mathscr{Q}'}(s_i)\}^{-1}$;*
> 3. *Generate o_i following the standard potential outcome model: $o_i = t_i o_i^{(1)} + (1 - t_i) o_i^{(0)}$, where $o_i^{(1)}$ and $o_i^{(0)}$ are given by (8).*

The proof of Lemma 1 is in Appendix A.1. Lemma 1 clarifies that the target function $\Delta(\cdot)$ is CATE from the perspective of causal data generation:

$$\Delta(q) = m(q) - \eta(q) = \mathbb{E}(o^{(1)} - o^{(0)} \mid q).$$

The causal identification assumptions such as consistency and unconfoundedness are satisfied under the Causal Data Generation Process. However, the positivity assumption on the propensity score, i.e., $p(s) \in (\epsilon, 1-\epsilon)$ for some constant $\epsilon > 0$, may be violated when the supports of $\mathscr{Q}$ and $\mathscr{Q}'$ are not fully overlapping. In particular, violation occurs if there exists a region of q such that $f_{\mathscr{Q}}(q) > 0$ while $f_{\mathscr{Q}'}(q) = 0$, or vice versa. In such cases, our proposed method remains valid after a data truncation step: we estimate $\Delta(q)$ only within the samples in the overlapped region of supports. We defer a detailed discussion of this truncation-based extension to future work in §5.

3.3 Causal meta-learning for $\Delta(q)$ and meta-router

Building on the seminal work of Künzel et al. (2019), many causal meta-learning approaches are developed, aiming to provide principled and flexible frameworks for CATE estimation. Meta-learners can incorporate any off-the-shelf ML algorithm, thereby offering substantial flexibility. Moreover, by leveraging ideas from orthogonal ML and semiparametric statistics (see, *e.g.*, Chernozhukov et al., 2018), meta-learners such as the R-learner (Nie & Wager, 2021) and the DR-learner (Kennedy, 2023) enjoy the oracle property. In particular, under mild conditions of nuisance function estimation, CATE meta-learners can be asymptotically equivalent to an oracle estimator that has access to the full set of individual treatment effects $\{o_i^{(1)} - o_i^{(0)}\}_{i=1}^n$, whereas in practice only one of $o_i^{(1)}$ or $o_i^{(0)}$ is observed for each i. This oracle property implies the statistical optimality of the R-learner and DR-learner for the estimation of $\Delta(q)$ in our setting; empirical studies also demonstrate their efficiency (Wu & Yang, 2022; Curth & Van der Schaar, 2021). In this paper, we focus on R- and DR-learners.

R-learner Let $\gamma(s) = \mathbb{E}(o \mid s)$ denote the marginal regression of the evaluation outcome on the query s, and let $p(s) = \Pr(t = 1 \mid s)$ denote the propensity score of receiving a GS evaluation. R-learner (Nie & Wager, 2021) constructs the orthogonalized residuals:

$$\tilde{o}_i = o_i - \hat{\gamma}(s_i), \quad \tilde{t}_i = t_i - \hat{p}(s_i),$$

where $\hat{\gamma}$ and $\hat{p}$ are any sensible sample-based estimators for γ and p. The R-learner then estimates $\Delta(\cdot)$ by solving the generalized least squares problem

$$\widehat{\Delta}_R(\cdot) = \arg\min_{h \in \mathcal{H}_\Delta} \frac{1}{n+m} \sum_{i=1}^{n+m} \big(\tilde{o}_i - \tilde{t}_i h(s_i)\big)^2 + \Lambda(h), \tag{9}$$

where $\mathcal{H}_\Delta$ is a pre-specified hypothesis space (*e.g.*, linear functions, random forests, or neural networks), and $\Lambda(h)$ is a regularizer to control complexity. This formulation is quasi-oracle efficient under mild conditions on nuisance estimators. Specifically, causal forests (Athey et al., 2019) is associated with the tree-based function class $\mathcal{H}_\Delta$ that can flexibly capture heterogeneous structures of $\Delta(\cdot)$ across different q.

DR-learner An alternative is the doubly robust (DR) learner of Kennedy (2023). It constructs a pseudo-outcome for each sample by combining outcome regression and propensity adjustment, thereby guaranteeing consistency if either component is correctly specified. Specifically, let $\mu_t(s) = \mathbb{E}(o \mid s, t)$ denote the conditional regression under treatment status $t \in \{0, 1\}$. Then, the DR pseudo-outcome is

$$\tilde{\phi}_i = \left(\frac{t_i - \hat{p}(s_i)}{\hat{p}(s_i)(1 - \hat{p}(s_i))} \right) (o_i - \hat{\mu}_{t_i}(s_i)) + \hat{\mu}_1(s_i) - \hat{\mu}_0(s_i).$$

The DR-learner estimates $\Delta(\cdot)$ by regressing ϕ_i on s_i:

$$\widehat{\Delta}_{DR}(\cdot) = \underset{h \in \mathcal{H}_\Delta}{\arg\min} \frac{1}{n+m} \sum_{i=1}^{n+m} \left(\tilde{\phi}_i - h(s_i) \right)^2 + \Lambda(h). \tag{10}$$

The doubly robust property ensures that $\widehat{\Delta}_{DR}(q)$ is consistent if either $\mu_t(\cdot)$ or $p(\cdot)$ is estimated consistently. Such a feature is particularly appealing in our setting, because the distributional discrepancy between $\mathcal{D}_G$ and $\mathcal{D}_P$ may induce misspecification in one nuisance model.

Both learners offer robustness against nuisance model misspecification and fit naturally into our integrative router. In this work, we consider both approaches as benchmark estimators for the shift function $\Delta(\cdot)$, and employ nonparametric ML regressors (*e.g.*, random forests, deep neural networks, and XGBoost) to capture heterogeneous structures of $\Delta(\cdot)$ across the query space.

The sample-splitting could be further employed into R- and DR-learners as discussed in (Nie & Wager, 2021; Kennedy, 2023) to avoid potential biases brought by nuisance function training through ML algorithms. We omit the details only for simplicity, and note that the sample splitting could be straightforwardly incorporated into our method. We refer interested readers to the aforementioned two papers and, *e.g.*, Chernozhukov et al. (2018) for further discussions.

Building on the construction of the oracle router in (6), we now replace the known shift function $\Delta(\cdot)$ with its meta-learner-based estimator $\hat{\Delta}(\cdot)$, and thereby formalize our two-step meta-router.

Meta-router

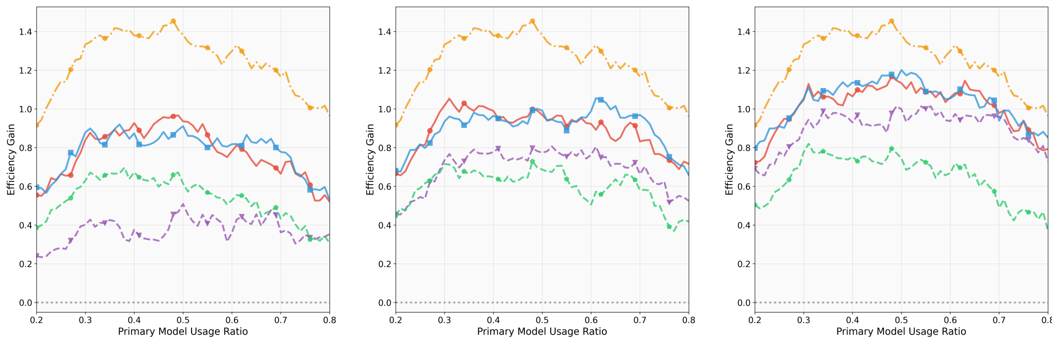
Inputs: $\mathcal{D} = \mathcal{D}_G \cup \mathcal{D}_P$; $\mathcal{H}_\Delta$, $\mathcal{H}_m$, $\Lambda(\cdot)$ specified by selected ML algorithms.

1. Estimate the shift function $\hat{\Delta}(\cdot)$ via certain CATE learning approaches, *e.g.*, the R-learner or DR-learner in (9) or (10) with nuisance functions trained over $\mathcal{D}$.
2. Meta-router $\hat{m}(\cdot) = \hat{m}_o(\cdot \mid \hat{\Delta})$ is obtained by solving (6) wherein $\Delta(\cdot)$ is replaced by $\hat{\Delta}(\cdot)$.

4 Numerical experiments on HealthBench

HealthBench (Arora et al., 2025) is a recently released benchmark designed to evaluate the performances of LLMs in open-ended healthcare scenarios. It consists 5000 professional user-model dialogues that were selected to span a wide range of healthcare scenarios. In total, 262 physicians across 26 specialties and 60 countries contributed to the creation of evaluation rubrics and consensus standards, make the evaluation mechanism precise in reflecting the qualities of LLM responses. The meta-evaluation verifies the trustworthy of these rubrics in faithfully reflecting physician judgement.

In our numerical experiments, we set Gemini 2.5 Pro as the primary model $\mathcal{M}_p$ (Comanici et al., 2025) and Gemma 3 12B as the alternative model $\mathcal{M}_a$ (Team et al., 2025), and collect their responses to all HealthBench questions. We then employ GPT-5-mini (OpenAI, 2025) for evaluation. For GS evaluations, each score-collecting prompt includes the evaluation rubrics, the original question, and the model response, and GPT-5-mini is asked to assign a score strictly following the rubrics. The score difference between $\mathcal{M}_p$ and $\mathcal{M}_a$ for each question is treated as the GS quality differences of two models. For preference-based evaluation, each prompt contains only the question and the two responses, and GPT-5-mini, asked to act as a medical expert, indicates whether $\mathcal{M}_p$ is better (1), comparable (0), or worse (-1), and this returned value is treated as the PB quality difference. We



(a) $n = 100$ (b) $n = 500$ (c) $n = 1000$

Figure 1: The efficiency gains of different routing strategies compared to the random routing baseline, against the primary model usage ratio. Subfigures correspond to varying GS sample sizes. Colors indicate different methods: oracle benchmark (yellow), meta-router via DR-learner (red), meta-router via R-learner (blue), predictive router using pooled GS–PB data (green), and predictive router using GS data only (purple). Query embeddings are reduced to dimension 50 via PCA, and all regressions are implemented using random forests.

normalize two types of quality gain evaluations to align their empirical variance (c.f., Remark 1(2)). We embed each query text to a 768-dimensional vector using the *gemini-embedding-001* model.

For each Monte Carlo (MC) round, we specify a machine learning algorithm $\mathcal{H}$, a GS sample size n, and a dimension d such that we further reduce the dimension of query text embedding to d via PCA. We then randomly split the data into three parts: a testing set $\mathcal{D}_{\text{text}}$ of with 500 queries and the corresponding GS evaluation outcomes r_i, a GS training set of size n, and a PB training set containing the remaining samples. Each training set only includes its corresponding type of evaluation outcomes. We compare six types of routers: (1) an oracle benchmark router that has access to the GS evaluation outcomes for all training queries in both GS and PB sets, and trains $m(q)$ via $\mathcal{H}$ using all these outcomes; (2) a predictive router that estimates $m(q)$ via $\mathcal{H}$ on the pooled GS and PB training data, without distinguishing evaluation types; (3) a predictive router that estimates $m(q)$ via $\mathcal{H}$ using only the GS training data; (4) a meta-router based on the R-learner trained on GS and PB data, with all involved predictions run by $\mathcal{H}$; (5) a meta-router based on the DR-learner trained on GS and PB data, with all involved predictions run by $\mathcal{H}$; (6) a random router that assigns each query to $\mathcal{M}_p$ with a fixed assignment probability.

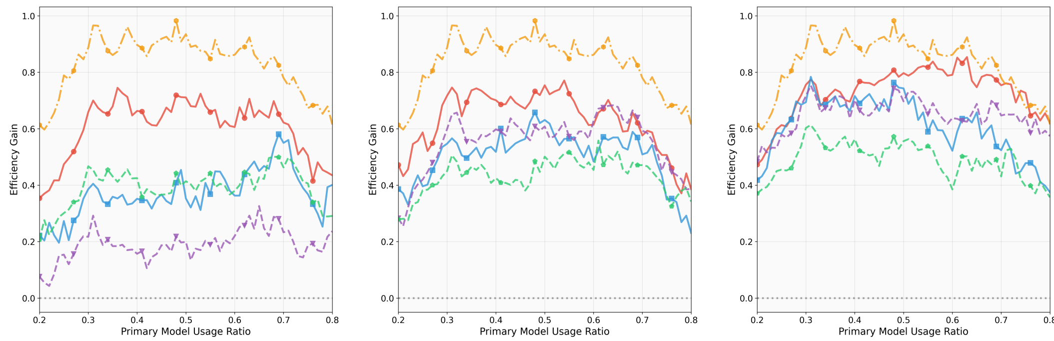
We consider two learning algorithms for $\mathcal{H}$: random forest (Breiman, 2001b) and XGBoost (Chen & Guestrin, 2016), three GS sample sizes $n \in \{100, 500, 1000\}$, and two PCA dimensions $d \in \{50, 100\}$. For each configuration and each Monte Carlo (MC) round, each router's decision rule follows (4), with $m(q)$ replaced by the corresponding estimator and binary cost functions as in (3). Given any weight w in (4), we compute the total efficiency (TE) of each router as

$$\text{TE} = \sum_{(q_i, r_i) \in \mathcal{D}_{\text{test}}} \mathbb{I}\{q_i \text{ is assigned to the primary model}\} \times r_i,$$

where r_i denotes the realized quality gain. By varying w, or equivalently the assignment probability for the random router, we obtain TE values under different primary model usage ratios (PMUR), defined as the proportion of queries assigned to the primary model among all testing samples. We run 100 MC rounds for each configuration and report the median TE across rounds for each router and PMUR level. To quantify relative performance, we further calculate the efficiency gain (EG) of a router as its improvement over the random router, averaging over 500 test samples:

$$\text{EG of any router} = \frac{\text{Median TE of any router} - \text{Median TE of the random router}}{500}.$$

The EGs of different routers versus PMURs under various settings are reported in Figure 1–2 and Figure 3–4 in the appendix. Our simulation results demonstrate the superior efficiency of meta-routers, particularly in imbalanced regimes with very limited GS data. In contrast, the predictive



(a) $n = 100$ (b) $n = 500$ (c) $n = 1000$

Figure 2: The efficiency gains of different routing strategies compared with the random routing baseline versus the primary model usage ratio. The query embedding dimension is reduced to 50 via PCA and all regressions are implemented via XGBoost. Other explanations are the same as Figure 1.

router trained on directly pooled GS and PB data, as considered in *e.g.*, Ong et al. (2024), shows little efficiency improvement even with relatively large GS sample sizes, highlighting the detrimental effect of bias $\Delta(q)$ in LLM routing. Among meta-routers, the R-learner-based variant consistently achieves the best performance across regression methods, dimensional settings, and different proportions of GS samples, further demonstrating the robustness of our proposed method after leveraging robust meta-learner frameworks.

5 Future work: truncation-based meta-router under positivity violation

Currently, our framework requires that the query distribution of GS data and that of the PB data share the common support. This requirement can be violated in practice when, *e.g.*, the GS data focuses on one category where responses can be easily justified, while the PB data are with regard to more subjective queries. When the supports of the two query distributions do not fully overlap, the positivity assumption for causal identification may be violated. A promising direction is to develop a truncation-based meta-router, which always incorporates all GS data but only retains preference data within the estimated overlap region of the two distributions. In particular, the overlap can be identified via efficient density ratio estimation. Then a meta-learner of $\Delta(\cdot)$ is trained only over the samples in $\mathcal{D}_G \cup \mathcal{D}_P$ which are considered as belonging to this region. Finally, when we train our truncation-based meta-router by solving (6) with obtained $\hat{\Delta}(\cdot)$ but only incorporating the samples $\mathcal{D}_P$ which belong to the detected overlap region. This truncation-based strategy offers a principled way to exploit abundant preference data while avoiding extrapolation bias outside the common support.

References

Rahul K Arora, Jason Wei, Rebecca Soskin Hicks, Preston Bowman, Joaquin Quiñonero-Candela, Foivos Tsimpourlas, Michael Sharman, Meghan Shah, Andrea Vallone, Alex Beutel, et al. Healthbench: Evaluating large language models towards improved human health. *arXiv preprint arXiv:2505.08775*, 2025.

Susan Athey, Julie Tibshirani, and Stefan Wager. Generalized random forests. 2019.

Ralph Allan Bradley and Milton E Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001a. doi: 10.1023/A:1010933404324.

Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001b.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 15(3):1–45, 2024.

Lingjiao Chen, Matei Zaharia, and James Zou. Frugalgpt: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*, 2023.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=cSimKw5p6R.

Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

Victor Chernozhukov, Denis Chetverikov, Mert Demirer, Esther Duflo, Christian Hansen, Whitney Newey, and James Robins. Double/debiased machine learning for treatment and structural parameters, 2018.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E Gonzalez, et al. Chatbot arena: An open platform for evaluating llms by human preference. In *Forty-first International Conference on Machine Learning*, 2024.

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025.

Alicia Curth and Mihaela Van der Schaar. Nonparametric estimation of heterogeneous treatment effects: From theory to learning algorithms. In *International Conference on Artificial Intelligence and Statistics*, pp. 1810–1818. PMLR, 2021.

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)*, pp. 4171–4186, 2019.

Dujian Ding, Ankur Mallick, Chi Wang, Robert Sim, Subhabrata Mukherjee, Victor Ruhle, Laks VS Lakshmanan, and Ahmed Hassan Awadallah. Hybrid llm: Cost-efficient and quality-aware query routing. *arXiv preprint arXiv:2404.14618*, 2024.

Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT Press, 2016.

Neel Guha, Julian Nyarko, Daniel Ho, Christopher Ré, Adam Chilton, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, et al. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Advances in neural information processing systems*, 36:44123–44279, 2023.

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

Qitian Jason Hu, Jacob Bieker, Xiuyu Li, Nan Jiang, Benjamin Keigwin, Gaurav Ranganath, Kurt Keutzer, and Shriyash Kaustubh Upadhyay. Routerbench: A benchmark for multi-llm routing system. *arXiv preprint arXiv:2403.12031*, 2024.

Guido W Imbens and Donald B Rubin. *Causal inference in statistics, social, and biomedical sciences*. Cambridge university press, 2015.

Edward H Kennedy. Towards optimal doubly robust estimation of heterogeneous causal effects. *Electronic Journal of Statistics*, 17(2):3008–3049, 2023.

Sören R Künzel, Jasjeet S Sekhon, Peter J Bickel, and Bin Yu. Metalearners for estimating heterogeneous treatment effects using machine learning. *Proceedings of the national academy of sciences*, 116(10):4156–4165, 2019.

Jaouad Moutrada, Stéphane Gaïffas, and Erwan Scornet. Minim minimax optimal rates for mondrian trees and forests. *The Annals of Statistics*, 48(4):2253–2276, 2020.

Xinkun Nie and Stefan Wager. Quasi-oracle estimation of heterogeneous treatment effects. *Biometrika*, 108(2):299–319, 2021.

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 arXiv:2406.18665*, 2024.

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

OpenAI. Gpt-5 technical report. https://openai.com/research/, 2025. Accessed: 2025-09-24.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, 2006. ISBN 978-0-262-18253-9. doi: 10.7551/mitpress/3206.001.0001. URL https://gaussianprocess.org/gpml/.

Donald B Rubin. Causal inference using potential outcomes: Design, Modeling, Decisions. *Journal of the American Statistical Association*, 100:322–331, 2005.

Johannes Schmidt-Hieber. Nonparametric regression using deep neural networks with relu activation function. *The Annals of Statistics*, 48(4):1875–1897, 2020.

Seamus Somerstep, Felipe Maia Polo, Allysson Flavio Melo de Oliveira, Prattyush Mangal, Mírian Silva, Onkar Bhardwaj, Mikhail Yurochkin, and Subha Maity. CARROT: A cost aware rate optimal router. In *ICLR 2025 Workshop on Foundation Models in the Wild*, 2025. URL https://openreview.net/forum?id=xEBOy2ze1U.

Dimitris Stripelis, Zhaozhuo Xu, Zijian Hu, Alay Dilipbhai Shah, Han Jin, Yuhang Yao, Jipeng Zhang, Tong Zhang, Salman Avestimehr, and Chaoyang He. Tensoropera router: A multi-model router for efficient llm inference. In *EMNLP (Industry Track)*, 2024.

Annalisa Szymanski, Noah Ziems, Heather A Eicher-Miller, Toby Jia-Jun Li, Meng Jiang, and Ronald A Metoyer. Limitations of the llm-as-a-judge approach for evaluating llm outputs in expert knowledge tasks. In *Proceedings of the 30th International Conference on Intelligent User Interfaces*, pp. 952–966, 2025.

Thomas Yu Chow Tam, Sonish Sivarajkumar, Sumit Kapoor, Alisa V Stolyar, Katelyn Polanska, Karleigh R McCarthy, Hunter Osterhoudt, Xizhi Wu, Shyam Visweswaran, Sunyang Fu, et al. A framework for human evaluation of large language models in healthcare derived from literature review. *NPJ digital medicine*, 7(1):258, 2024.

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

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996.

Andrey N. Tikhonov and Vasiliy Y. Arsenin. *Solutions of ill-posed problems*. V. H. Winston & Sons, Washington, D.C.: John Wiley & Sons, New York, 1977. Translated from the Russian, Preface by translation editor Fritz John, Scripta Series in Mathematics.

Asterios Tsiourvas, Wei Sun, and Georgia Perakis. Causal llm routing: End-to-end regret minimization from observational data. *arXiv preprint arXiv:2505.16037*, 2025.

Larry Wasserman. *All of nonparametric statistics*. Springer, 2006.

Koki Wataoka, Tsubasa Takahashi, and Ryokan Ri. Self-preference bias in llm-as-a-judge. *arXiv preprint arXiv:2410.21819*, 2024.

Lili Wu and Shu Yang. Integrative r-learner of heterogeneous treatment effects combining experimental and observational studies. In *Conference on Causal Learning and Reasoning*, pp. 904–926. PMLR, 2022.

Tuo Zhang, Asal Mehradfar, Dimitrios Dimitriadis, and Salman Avestimehr. Leveraging uncertainty estimation for efficient llm routing. *arXiv preprint arXiv:2502.11021*, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023a.

Zangwei Zheng, Xiaozhe Ren, Fuzhao Xue, Yang Luo, Xin Jiang, and Yang You. Response length perception and sequence scheduling: An llm-empowered llm inference pipeline. *Advances in Neural Information Processing Systems*, 36:65517–65530, 2023b.

Lianghui Zhu, Xinggang Wang, and Xinlong Wang. Judgelm: Fine-tuned large language models are scalable judges. *arXiv preprint arXiv:2310.17631*, 2023.

A Appendix

A.1 Proof of Lemma 1

The density function of (s, t, o) in GS–PB joint Data Generation Process could be written as

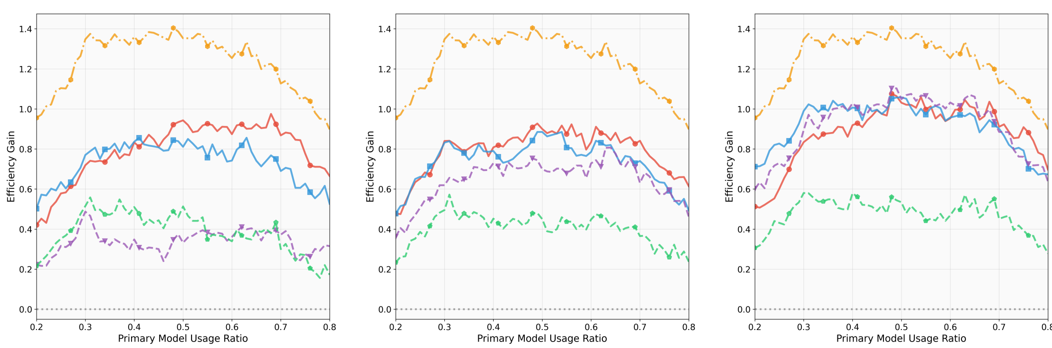
$$f(s, t, o) = \kappa^t (1-\kappa)^{1-t} f_{\mathscr{Q}}^t(s) f_{\mathscr{Q}'}^{1-t}(s) f_r^t(o \mid s) f_y^{1-t}(o \mid s),$$

where $f_r(\cdot \mid s)$ and $f_y(\cdot \mid s)$ represent the conditional probability density function of r_i and y_i given $q_i = s$, following (1) and (2), respectively. This could be further written as

$$f(s, t, o) = \underbrace{(\kappa f_{\mathscr{Q}}(s) + (1-\kappa) f_{\mathscr{Q}'}(s))}_{f_{\kappa \mathscr{Q} + (1-\kappa)\mathscr{Q}'}(s)} \cdot \underbrace{\frac{\kappa^t (1-\kappa)^{1-t} f_{\mathscr{Q}}^t(s) f_{\mathscr{Q}'}^{1-t}(s)}{\kappa f_{\mathscr{Q}}(s) + (1-\kappa) f_{\mathscr{Q}'}(s)}}_{Pr(t_i = t \mid s) = tp(s) + (1-t)p(s)} \cdot \underbrace{f_r^t(o \mid s) f_y^{1-t}(o \mid s)}_{f_{o(t)}(o \mid s)}, \quad (11)$$

recalling the notation in Causal Data Generation Process, and thereby show the distributional equivalence of two processes. □

A.2 Additional numerical results for §4



(a) $n = 100$ (b) $n = 500$ (c) $n = 1000$

Figure 3: The efficiency gains of different routing strategies compared with the random routing baseline versus the primary model usage ratio. The query embedding dimension is reduced to 100 via PCA and all regressions are implemented via random forest. Other explanations are the same as Figure 1.

(a) $n = 100$ (b) $n = 500$ (c) $n = 1000$

Figure 4: The efficiency gains of different routing strategies compared with the random routing baseline versus the primary model usage ratio. The query embedding dimension is reduced to 100 via PCA and all regressions are implemented via XGBoost. Other explanations are the same as Figure 1.

A.3 The Use of Large Language Models (LLM)

For this project, LLMs were used to polish the writing of the main paper and to assist with coding for the numerical experiments.