RouteLMT: Learned Sample Routing for Hybrid LLM Translation Deployment

Yingfeng Luo[1], Hongyu Liu[1], Dingyang Lin[1], Kaiyan Chang[1], Chenglong Wang[1]
Bei Li[1], Quan Du[2], Tong Xiao[1,2]*, Jingbo Zhu[1,2],
[1] School of Computer Science and Engineering, Northeastern University, Shenyang, China
[2] NiuTrans Research, Shenyang, China
luoyingfeng_neu@outlook.com
{xiaotong,zhujingbo}@mail.neu.edu.cn

Abstract

Large Language Models (LLMs) have achieved remarkable performance in Machine Translation (MT), but deploying them at scale remains prohibitively expensive. A widely adopted remedy is the hybrid system paradigm, which balances cost and quality by serving most requests with a small model and selectively routing a fraction to a large model. However, existing routing strategies often rely on heuristics, external predictors, or absolute quality estimation, which fail to capture whether the large model actually provides a worthwhile improvement over the small one. In this paper, we formulate routing as a budget allocation problem and identify marginal gain, i.e., the large model's improvement over the small model, as the optimal signal for budgeted decisions. Building on this, we propose **RouteLMT** (routing for LLM-based MT), an efficient in-model router that predicts this expected gain by probing the small translator's prompt-token representation, without requiring external models or hypothesis decoding. Extensive experiments demonstrate that our RouteLMT outperforms heuristics, quality/difficulty estimation baselines, achieving a superior quality–budget Pareto frontier. Furthermore, we analyze regression risks and show that a simple guarded variant can mitigate severe quality losses.

1 Introduction

Large language models (LLMs) have recently demonstrated strong translation capability across diverse domains and language pairs (Yang et al., 2023; Alves et al., 2024; Xu et al., 2024; Cui et al., 2025; Luo et al., 2025a). However, deploying a single large model for all production requests at scale is often impractical in industrial translation systems due to strict constraints on serving cost, tail latency, and compute capacity. A common strategy is therefore hybrid deployment, where a smaller and cheaper model serves the majority of traffic, while a larger and more capable model is reserved for inputs where the small model is likely to translate poorly, or where the large model is expected to yield the largest quality improvements.

Method	Decision timing	Needs hypothesis?	Extra model
Hendy et al. (2023)	Post-route	✓	✓
Wu et al. (2025)	Pre-route	✗	✓
Farinhas et al. (2025)	Post-route	✓	✓
Proietti et al. (2025)	Pre-route	✗	✓
RouteLMT (Ours)	Pre-route	✗	✗

Table 1: Comparison of representative hybrid MT routing approaches.

Such a hybrid deployment introduces a key operational question: given a fixed large model call budget, which inputs should trigger the large model? Naive routing, such as simple heuristics (e.g., length), can waste large model capacity on easy inputs and miss high-gain cases where the large model delivers the most significant boost. Recent work has begun to study this question (see Table 1), including pre-route deciders based on source features to reduce unnecessary large model calls (Wu et al., 2025; Proietti et al., 2025) and post-route, QE-based deferral strategies that first generate a small-model hypothesis translation and then use QE to defer low-quality cases to a larger model (Hendy et al., 2023; Farinhas et al., 2025).

Despite these advances, it remains challenging to design routers that are both accurate and lightweight. Most existing approaches either (i) rely on separate external routers (e.g., a source-feature classifier or QE model), or (ii) require generating a hypothesis from the small model before making the decision. Both choices have clear limitations. External routers increase operational complexity while overlooking the rich representations of the small translator, which may encode cues

* Corresponding author.

about input difficulty and expected translation quality. Meanwhile, hypothesis-dependent strategies require an extra decoding and scoring step, increasing computation and latency. Beyond overhead, many routers optimize proxies such as input difficulty or absolute quality, which are misaligned with a budgeted routing objective. For example, hard inputs may be hard for both models, yielding zero gain, while some seemingly simple inputs might still benefit substantially from the large model.

In this work, we formulate routing as a budget allocation problem: maximizing overall translation quality under a fixed budget of large-model calls. This formulation motivates ranking requests by the expected *marginal gain* of upgrading (the large model's improvement over the small model) and allocating the budget on the top-ranked instances. To predict this gain with minimal overhead, we avoid hypothesis-dependent features and instead embed the router into the small translator. Concretely, motivated by recent findings on the informativeness of final-token representations (Zhu et al., 2025; Lee et al., 2025; Dong et al., 2025), we predict gain directly from the small translator's hidden state at the last token of the translation prompt via a simple regression head. We jointly train this head with the small translator using parameter-efficient LoRA adaptation (Hu et al., 2022), yielding an in-model, hypothesis-free, and translation direction-aware router without deploying a separate QE model.

To assess the efficacy of different routing policies, we systematically compare routing strategies in a budgeted hybrid LLM translation setting, spanning heuristics, learned quality/difficulty prediction, and learned gain prediction. We evaluate routers with ranking- and allocation-oriented metrics, and summarize quality–cost trade-offs using Pareto curves that characterize translation quality as a function of the routed-to-large fraction. Across multiple directions and budgets, we find that gain-based in-model routing delivers the best quality–cost frontier among all routers we evaluate.

In summary, our contributions are threefold:

- We formulate hybrid LLM MT routing as a budget allocation problem and derive expected *marginal gain* as the key signal for routing decisions.
- We introduce **RouteLMT**, an in-model, hypothesis-free, direction-aware router that leverages the small translator's internal representations to predict marginal gain, without requiring external models or hypothesis decoding.
- We empirically validate gain-based in-model routing, showing consistent improvements over heuristics and quality/difficulty-based learned routing methods.

2 Related Work

Hybrid and adaptive inference is a pragmatic way to balance quality and resource usage. This idea has recently regained attention with LLM deployments, where the gap between model quality and serving cost/latency can be substantial.

Model routing and LLM cascades. Several works study how to route requests among multiple LLMs to improve the quality–cost frontier. FrugalGPT (Chen et al., 2024), Hybrid LLM (Ding et al., 2024), and MixLLM (Wang et al., 2025) adopt an LLM cascade perspective and study policies that selectively invoke stronger models to reduce cost while maintaining accuracy. RouteLLM (Ong et al., 2024) learns routers from preference data to choose between a strong and a weak LLM at inference time. Gupta et al. (2024) systematically studies uncertainty-based deferral for generative tasks, showing that naive sequence-probability uncertainty suffers from length bias and that learned deferral rules leveraging token-level uncertainty can yield better cost–quality trade-offs.

Hybrid translation and quality-aware deferral. In machine translation, hybrid deployment has appeared in multiple forms. Recent work combines NMT and LLM translation and learns a source-feature-based decider to reduce unnecessary LLM calls in hybrid systems (Wu et al., 2025). Another prominent direction is quality-aware deferral: the system first runs a smaller translator and then uses a QE signal to decide whether to defer to a larger model. Farinhas et al. (2025) shows that QE-based deferral can match large-model translation quality while invoking it for only a fraction of examples. Relatedly, Hendy et al. (2023) propose a QE-threshold approach that triggers a second (stronger) translation when the initial output is predicted to be low quality. A complementary line of work builds predictors that estimate MT difficulty from the source sentence to identify challenging inputs (Don-Yehiya et al., 2022; Proietti et al., 2025).

Our work focuses on budgeted sample routing for hybrid LLM translation, but differs from prior

MT routing in how routing signals are obtained and system design. Rather than relying on a separately served router or post-decoding QE signals, we estimate marginal gain directly from the small translator's internal prompt representations, without requiring a decoded hypothesis.

3 Problem Formulation

We consider sample routing for a hybrid machine translation system with a fixed budget of large-model calls. Each request is a source sentence x paired with a translation direction $d \in \mathcal{D}$ (e.g., En→Zh). The system has access to two translation models: a low-cost *small* model M_s and a higher-cost *large* model M_ℓ. For a request (x, d), the two models produce translations

$$\hat{y}_s = M_s(x; d), \qquad \hat{y}_\ell = M_\ell(x; d). \tag{1}$$

A router chooses which model serves the request, resulting in the deployed output $\hat{y}$.

Let $\Phi(x, \hat{y}, y^\star) \in \mathbb{R}$ denote a reference-based translation quality score, where $y^\star$ is the human reference translation [1]. Define the per-request quality under each model:

$$\begin{aligned} q_s(x; d) &= \Phi(x, \hat{y}_s, y^\star)\,, \\ q_\ell(x; d) &= \Phi(x, \hat{y}_\ell, y^\star)\,. \end{aligned} \tag{2}$$

We study *budgeted* routing, where only a fraction $p \in (0, 1]$ of requests may be served by M_ℓ. Let $z(x, d) \in \{0, 1\}$ be the routing decision, where $z = 1$ indicates routing to M_ℓ and $z = 0$ indicates routing to M_s. The budget constraint is

$$\mathbb{E}[z(x, d)] \leq p, \tag{3}$$

with expectation taken over the request distribution.

Under this constraint, the objective is to maximize the expected system output quality:

$$\begin{aligned} &\max_R \mathbb{E}\Big[z(x, d) q_\ell(x; d) + (1 - z(x, d)) q_s(x; d)\Big] \\ &\text{s.t. } \mathbb{E}[z(x, d)] \leq p, \end{aligned} \tag{4}$$

where R denotes the routing policy that maps (x, d) to z. It is useful to rewrite (4) by defining the *marginal gain* of using the large model over the small model:

$$g(x; d) = q_\ell(x; d) - q_s(x; d). \tag{5}$$

[1] We use a reference-based Φ to obtain a less noisy supervision signal for isolating algorithmic gains under controlled conditions. The framework is agnostic to how Φ is obtained, and can substitute $\Phi(x, \hat{y})$ from a reference-free QE model.

Substituting (5) into (4) yields

$$\begin{aligned} \mathbb{E}\Big[z\, q_\ell + (1 - z)\, q_s\Big] &= \mathbb{E}\big[q_s(x; d)\big] \\ &\quad + \mathbb{E}\big[z(x, d)\, g(x; d)\big]. \end{aligned} \tag{6}$$

Since $\mathbb{E}[q_s(x; d)]$ is constant with respect to the router, maximizing expected system quality under a fixed call budget reduces to maximizing $\mathbb{E}[z(x, d)\, g(x; d)]$. Therefore, the router should allocate the limited large-model budget to requests with the largest gains $g(x; d)$, i.e., prioritizing inputs where M_ℓ is expected to improve most over M_s. This motivates routing policies that rank requests by an estimate of g and allocate the budget to the top-ranked ones. This also indicates that intuitive proxies such as difficulty or the small model's absolute quality can be misaligned with the budgeted routing objective, as challenging inputs or low-quality small-model outputs do not necessarily imply a large improvement from the large model.

4 Method

Motivated by the objective in Equation (6), we learn a gain predictor that estimates the marginal gain $g(x; d) = q_\ell(x; d) - q_s(x; d)$ from the input, and use it to allocate a fixed large-model budget.

In -model gain router. Let $P(x, d)$ denote the translation prompt constructed from the source sentence x and direction d. Given request (x, d), we run a single prefill step of the small translator M_s on $P(x, d)$ and extract the hidden representation of the final prompt token, denoted $h(x; d)$. To predict the gain, we project $h(x; d)$ to a scalar score via a lightweight linear head f_θ:

$$\hat{g}(x; d) = f_\theta(h(x; d))\,, \tag{7}$$

We employ LoRA (Hu et al., 2022) to adapt the small translator M_s and jointly train the regression head f_θ. Furthermore, because the translation direction d is embedded in the prompt, the representation $h(x; d)$ is inherently direction-conditioned, enabling a single router to learn direction-aware policies across multiple language pairs.

Supervision and training. For each instance $(x, y^\star, d)$, we obtain translations from M_s and M_ℓ and compute the gain label $g(x; d)$ as the quality difference between their Φ scores. We train the router with an MSE loss:

$$\mathcal{L}_{\text{gain}} = \mathbb{E}_{(x, y^\star, d)}\Big[\big(\hat{g}(x; d) - g(x; d)\big)^2\Big]\,. \tag{8}$$

Inference-time routing policy. Under a large-model budget p, we route requests with the highest predicted gain $\hat{g}(x; d)$. For offline evaluation, we select the top-p fraction of examples by $\hat{g}(x; d)$. For streaming deployment, this can be implemented by applying a threshold τ_p on $\hat{g}(x; d)$, calibrated on held-out traffic, so that approximately a fraction p of incoming requests satisfy $\hat{g}(x; d) \geq \tau_p$.

5 Experiments

5.1 Data and Models

We evaluate four translation directions: En↔Zh and En↔Ru. For the training, we use the *General Translation* split of ComMT (Luo et al., 2025b), a compiled dataset aggregating high-quality parallel corpora from multiple sources. For evaluation, we combine FLORES-200 devtest (Costa-jussà et al., 2022), WMT24++ (Deutsch et al., 2025), and BOUQuET (Andrews et al., 2025). Detailed statistics are provided in Table 6.

We employ the recently released open LMT-60 multilingual MT model family (Luo et al., 2025a), which provides models across a range of sizes. We choose LMT-60-0.6B as M_s and LMT-60-8B as M_ℓ as a representative small/large configuration for hybrid routing. We apply LoRA to all linear layers of M_s with rank r=8 and α=32; full hyperparameter settings are in Appendix Table 5.

5.2 Baselines

We compare RouteLMT against random routing, heuristic routers, and learned routers. Given a large-model budget p, each method assigns a routing score to each request and routes approximately the top-p fraction to M_ℓ (or bottom-p when lower scores indicate higher priority).

Heuristic routers include **Length**, **Rarity**, and **Entropy**. Learned routers are grouped by router type and learning target. In terms of router type, we distinguish between (i) in-model routers that leverage the small translator's internal representations and (ii) external routers that rely on a separate model operating on the source sentence. In terms of learning target, we consider predicting marginal gain (Δ) versus small-model quality (Q). We denote our proposed in-model gain router as **RouteLMT** (i.e., RouteLMT-Δ), and its quality variant as **RouteLMT-Q**. As external counterparts, we train **XLM-R-Δ** and **XLM-R-Q** using XLM-RoBERTa-L (Conneau et al., 2020) on the same data as RouteLMT to predict Δ and Q from the source sentence. Additionally, we include **sentinel-src-24/25** (Proietti et al., 2025), a strong baseline that models translation difficulty via quality estimation. A summary is provided in Table 7, with full details in Appendix A.

5.3 Metrics

We adopt XCOMET-XXL (Guerreiro et al., 2024) as the reference-based translation quality evaluator Φ, which has been shown to achieve high agreement with human judgments. We evaluate routing performance using three complementary metrics:

- **Spearman**: The rank correlation between predicted routing scores and the oracle marginal gain. This assesses the router's ability to induce a globally consistent ranking of improvement potential.
- **HitRate@p**: The overlap fraction between the router-selected top-p set and the oracle top-p set. This evaluates selection accuracy (or decision accuracy) under a budget p.
- **MeanΔ@p**: The average marginal gain of the router-selected top-p examples. This measures the actual benefit of the routed subset.

6 Results and Analyses

6.1 Router Performance

Table 2 reports router performance under a fixed large-model budget p=0.3, which we use as a representative operating point for practical quality–cost trade-off. We include two oracle baselines to contextualize the upper bounds. **Gain Oracle** routes by the true marginal gain $g(x; d)$ and is therefore optimal under the budget constraint, whereas **Quality Oracle** routes by the true small-model quality, which requires first decoding the small-model hypothesis and then scoring it, and thus serves as an upper bound for quality-based strategy. A key takeaway from the oracles is that maximizing absolute quality is not equivalent to maximizing marginal gain: even with oracle access, quality-based routing is only moderately correlated with true gain (Spearman: 0.67) and consequently underperforms the Gain Oracle in both HitRate and MeanΔ.

Among practical routers, RouteLMT (in-model gain prediction) performs best overall across all three metrics. It achieves the highest Spearman correlation and HitRate@p, indicating a consistently better global ranking and top-p selection capabilities than compared methods. More importantly,

Method	Spearman ↑	HitRate@p ↑					MeanΔ@p ↑				
	Avg.	En→Zh	En→Ru	Zh→En	Ru→En	Avg.	En→Zh	En→Ru	Zh→En	Ru→En	Avg.
Gain Oracle	1.00	100.00	100.00	100.00	100.00	100.00	17.12	29.36	10.81	20.64	19.48
Quality Oracle	0.67	74.26	75.94	73.96	76.24	75.10	14.51	26.33	8.40	17.68	16.73
Random	0.00	30.00	30.00	30.00	30.00	30.00	4.90	10.88	2.25	5.28	5.83
Length	0.24	47.43	48.32	47.03	42.77	46.39	8.50	17.41	3.76	7.72	9.35
Rarity	0.14	35.64	33.86	42.28	41.68	38.37	6.16	12.00	3.62	8.47	7.56
Entropy	0.09	36.93	38.61	38.61	34.85	37.25	6.05	13.29	3.55	6.89	7.45
sentinel-src-24	0.34	54.36	55.45	54.95	55.25	55.00	9.93	19.49	4.83	10.83	11.27
sentinel-src-25	0.31	50.89	52.57	51.49	51.49	51.61	9.36	18.38	4.57	10.13	10.61
XLM-R-Δ	0.32	54.06	55.84	51.68	52.77	53.59	9.72	19.49	4.59	10.29	11.02
XLM-R-Q	0.28	50.59	51.78	50.30	48.22	50.22	9.18	18.27	4.40	9.22	10.27
RouteLMT-Q	0.37	54.95	59.21	54.95	55.05	56.04	9.90	20.47	5.00	11.70	11.77
RouteLMT	**0.40**	**56.24**	**60.50**	**55.64**	**56.93**	**57.33**	**10.24**	**21.03**	**5.12**	**12.13**	**12.13**

Table 2: Router performance across four translation directions. Spearman is computed over the full set to measure global ranking quality, while HitRate@p and MeanΔ@p are evaluated under a fixed budget p=0.3 (30% routed to the large model). **Bold** numbers indicate the best score, and underlined numbers the second best.

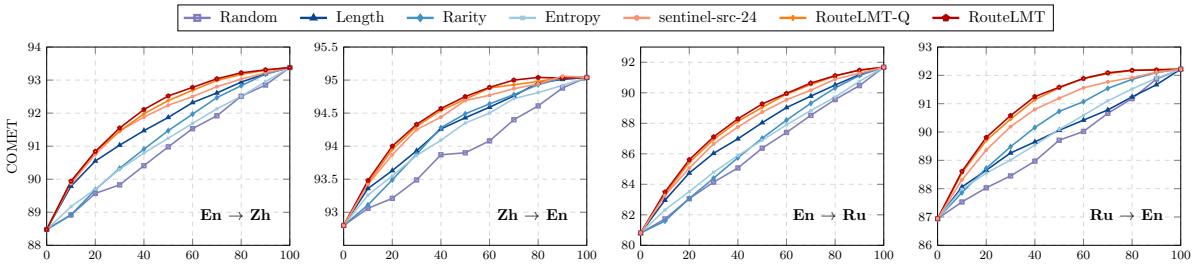

Figure 1: Quality–budget trade-offs of hybrid translation routing. We sweep the large-model budget p (route-to-large rate) and report the quality of the resulting hybrid system. Higher curves indicate a better Pareto frontier.

it yields the largest MeanΔ@p (12.13), improving over the strongest heuristic Length baseline (9.35) by +2.78 and more than doubling the Random baseline (5.83). When comparing learning objectives, RouteLMT consistently surpasses its quality-predicting counterpart (RouteLMT-Q), confirming that directly modeling marginal gain aligns better with the budgeted allocation goal. Furthermore, external routers (XLM-R-Δ/Q and sentinel-src-24/25) consistently trail the in-model predictors (RouteLMT and RouteLMT-Q), highlighting the value of leveraging the small translator's internal representations for learning routing policies. We also observe these trends generalize to out-of-domain settings (medical and colloquial); detailed results are reported in Appendix Section B.

Overall, these results show that effective routing benefits from signals encoded in the small translator's prompt representations, and that predicting marginal gain is more effective than relying on quality/difficulty proxies. Despite these improvements, the remaining gap to the **Gain Oracle** highlights substantial headroom for future improvements in learned routing.

6.2 Quality–Budget Pareto Frontier

To characterize router behavior beyond a single operating point, we sweep the large-model budget p and visualize the resulting quality–budget curves in Figure 1. For clarity, we plot representative policies from each family (random, heuristics, and learned routers), and include sentinel-src-24 as the strongest external predictor.

Figure 1 shows that learned routers (red family) consistently dominate heuristic policies (blue family) across directions. This indicates that routing benefits from learned decision signals beyond surface heuristics such as length- or rarity-based proxies. A secondary pattern is diminishing returns as p increases: once the highest-return requests have been routed, additional large-model calls yield smaller quality gains, so the curves gradually converge near full routing.

Among learned routers, RouteLMT achieves the best quality–budget Pareto frontier across directions over most budgets. In addition, RouteLMT-Q performs better than the external predictor sentinel-src-24, indicating that leveraging the internal representations of the translation backbone provides a

more informative routing signal than a stand-alone model operating solely on source text.

6.3 Risk Analysis

In deployment, improving the average is not enough; practitioners also care about regression risk and their operational impact. We therefore analyze the gain distribution of routed-to-large requests under the same budget (p=0.3). For each routed example, we compute $g = \Phi(\hat{y}_\ell) - \Phi(\hat{y}_s)$ and bucket it into Severe loss ($g \leq -5$), Minor loss ($-5 < g < -0.5$), Tie ($|g| \leq 0.5$), and Substantial gain ($g > 0.5$). Figure 2 reports bucket proportions averaged across four directions (per-direction results in Appendix Figure 3).

Figure 2 shows that learned routing reallocates the large-model budget toward clearly beneficial upgrades. Compared to random and simple heuristics, learned routers markedly reduce *Tie* cases, where large-model calls yield little benefit, and correspondingly increase the share of *Substantial gain* invocations. Among practical methods, RouteLMT achieves the highest substantial-gain proportion while keeping minor-loss rates low, indicating a more efficient use of the fixed budget. Notably, severe-loss rates remain non-trivial and are similar across most non-random policies (roughly 8–9%), suggesting that the main gains of learned routing come from prioritizing high-return requests rather than eliminating extreme negative-gain cases.

6.4 Guarded Routing for Risk Control

To further mitigate severe regressions, we explore a simple guarded variant that augments gain-based routing with an additional quality filter. Concretely, we first rank requests by predicted gain (RouteLMT) and then apply a quality guard, while keeping the overall route-to-large rate fixed at p=0.3 to ensure a fair comparison. We set the guard threshold using a fixed quantile (30% in our experiments) as a simple calibration choice. We consider two guards: Quality (predict) uses an in-model quality predictor (RouteLMT-Q), while Quality (hypo) first decodes the M_s hypothesis and then applies a quality scorer.

Table 3 shows that Quality (predict) has little effect in this setting, yielding nearly identical severe-loss and MeanΔ@p to gain-only routing. In contrast, Quality (hypo) substantially reduces severe loss (from 8.19% to 5.69%) and improves MeanΔ@p (from 12.13 to 16.73), suggesting that a portion of catastrophic cases can be detected given a post-route verifier.

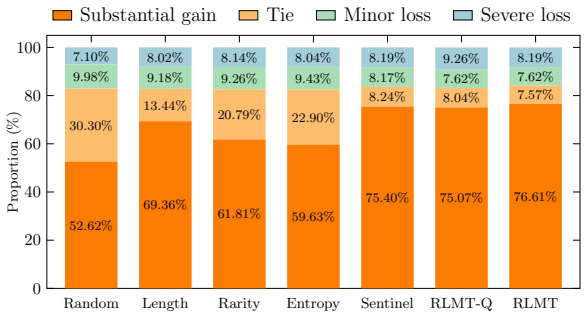

Figure 2: Gain-bucket distribution among routed-to-large model requests under budget p=0.3, averaged across four directions.

Method	Severe loss ↓	MeanΔ@p ↑
Random	7.10%	5.83
Gain	8.19%	12.13
Gain + Quality (predict)	8.19%	12.24
Gain + Quality (hypo)	5.69%	16.73

Table 3: Guarded routing (p=0.3) with gain ranking and quality guards.

Overall, these results suggest a coarse-to-fine scheme: use gain-based pre-routing to handle most requests with low overhead, and optionally apply a post-route check on the remaining candidates when stricter risk control is needed, at the cost of additional decoding and scoring.

6.5 Case Study

Table 4 provides illustrative examples showing both large positive gains and negative-gain regressions when upgrading to M_ℓ. These cases offer qualitative context for the regression patterns discussed in §6.3.

In the positive-gain examples, the large model mainly corrects systematic weaknesses of the small translator. For En→Zh, M_s occasionally fails on code-switching and leaves English fragments untranslated (e.g., "halfway"), whereas M_ℓ produces a complete and fluent Chinese rendering ($\Delta = +25.36$). We also observe that M_ℓ better handles figurative or non-literal expressions: when M_s translates a metaphor too literally, M_ℓ produces a more idiomatic formulation with improved adequacy and fluency ($\Delta = +25.42$).

The negative-gain examples highlight plausible regression modes. One involves abbreviation handling. In the headline example, M_ℓ attempts to interpret "WA" and expands it to a specific entity, but chooses the wrong one ("Western Aus-

Case	Type	Item	Text	Score
En→Zh $\Delta = +25.36$	Code -switch	Source	I'm about halfway thru the first study unit	–
		Reference	第一个学习单元我已经学完一半了	–
		Small	我大约 halfway 到了第一个学习单元	74.59
		Large	我已经完成了第一个学习单元的一半内容	99.95
En→Zh $\Delta = +25.42$	Literal	Source	we understand that it is hard to get to these positions of wealth and power without stepping on others along the way.	–
		Reference	我们逐渐意识到，一个人如果想要获得财富和权力，就很难保证自己不会去伤害别人、利用别人。	–
		Small	我们明白，要想获得财富和权力的这些地位，就必须在途中 踩到别人的脚 。	72.86
		Large	我们知道，如果不踩着别人往上爬，就很难获得财富和权力。	98.28
En→Zh $\Delta = -10.10$	Entity	Source	Bring back oversight for WA's jails. Lives depend on it	–
		Reference	恢复对华盛顿州看守所的监管刻不容缓，人命关天！	–
		Small	恢复 WA 监狱的监管。生命取决于此	83.60
		Large	恢复对 西澳大利亚州 监狱的监管。生命攸关	73.50
Zh→En $\Delta = -36.33$	Over-paraphrase	Source	我喜欢说什么就说什么。	–
		Reference	I can say whatever I want.	–
		Small	I like to say whatever I want to say.	93.52
		Large	I like to speak my mind .	57.19

Table 4: Case study examples with highlighted error spans. $\Delta = q_L - q_S$ is computed from XCOMET-XXL scores (scaled to 0–100). Type summarizes the primary error type.

tralia"), leading to a clear adequacy error. By contrast, M_s largely preserves "WA" as-is, which is less informative but avoids committing to an incorrect expansion; under reference-based scoring, the large model is penalized more heavily for the wrong disambiguation ($\Delta = -10.10$). A second class involves semantic drift under paraphrasing. For Zh→En, M_ℓ generates a more idiomatic paraphrase ("speak my mind") that shifts meaning relative to the intended sense ("say whatever I want"), resulting in a large negative gain ($\Delta = -36.33$). Although reference-based metrics can sometimes penalize benign paraphrases, here the large-model output also shifts the intended meaning, so the regression is not merely stylistic.

Taken together, these cases suggest that many severe regressions arise from rare but high-impact generation errors (e.g., wrong entity resolution or meaning drift) rather than from the same "difficulty" factors that make M_s weak. This helps explain why severe-loss rates can remain similar across many routing policies.

7 Conclusion

In this paper, we studied learned sample routing for hybrid LLM translation deployment. By formulating routing as a budget allocation problem, we identify marginal gain as the key decision signal and propose RouteLMT, an in-model router built on the small translator that predicts gain from its translation-prompt representations using lightweight LoRA adaptation. Across four translation directions, gain-based in-model routing achieves the strongest quality–budget trade-off among evaluated baselines, outperforming heuristic policies and separately trained routers that operate solely on source text. We further analyzed gain-bucket distributions to characterize regression risk, and showed that a simple guarded variant can reduce severe-loss cases. Overall, our results suggest that probing small-model representations offers an effective approach to budgeted hybrid translation routing.

Limitations

Our study provides an initial validation of the approach, and we note several limitations. First, the gain supervision is derived from an automatic reference-based quality metric, which may not fully reflect human judgments or application-specific utility, and the learned router can inherit its biases. Second, we focus on a two-model hybrid setting with a fixed route-to-large budget; more complex cascades, multi-tier budgets, and latency-aware objectives remain to be explored. In addition, our experiments use a specific small/large pair (0.6B

vs. 8B), a representative small/large configuration; how the same routing design behaves with other model families, larger scales, or different capability gaps remains to be validated. Finally, our experiments cover four directions (En↔Zh and En↔Ru); broader language coverage and large-scale multilingual settings, where routing behavior may vary across scripts and resource levels, are important directions for future work.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (Nos. U24A20334 and 62276056), the Yunnan Fundamental Research Projects (No.202401BC070021), the Yunnan Science and Technology Major Project (No. 202502AD080014), the Fundamental Research Funds for the Central Universities (Nos. N25BSS054 and N25BSS094), and the Program of Introducing Talents of Discipline to Universities, Plan 111 (No.B16009).

References

Duarte M. Alves, José Pombal, Nuno Miguel Guerreiro, Pedro Henrique Martins, João Alves, M. Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, Pierre Colombo, José G. C. de Souza, and André F. T. Martins. 2024. Tower: An open multilingual large language model for translation-related tasks. *CoRR*, abs/2402.17733.

Pierre Andrews, Mikel Artetxe, Mariano Coria Meglioli, Marta R Costa-jussà, Joe Chuang, David Dale, Mark Duppenthaler, Nathanial Paul Ekberg, Cynthia Gao, Daniel Edward Licht, and 1 others. 2025. Bouquet: dataset, benchmark and open initiative for universal quality evaluation in translation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 27503–27523.

Lingjiao Chen, Matei Zaharia, and James Zou. 2024. Frugalgpt: How to use large language models while reducing cost and improving performance. *Trans. Mach. Learn. Res.*, 2024.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 8440–8451.

Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Y. Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loïc Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, and 19 others. 2022. No language left behind: Scaling human-centered machine translation. *CoRR*, abs/2207.04672.

Menglong Cui, Pengzhi Gao, Wei Liu, Jian Luan, and Bin Wang. 2025. Multilingual machine translation with open large language models at practical scale: An empirical study. In *NAACL (Long Papers)*, pages 5420–5443. Association for Computational Linguistics.

Daniel Deutsch, Eleftheria Briakou, Isaac Caswell, Mara Finkelstein, Rebecca Galor, Juraj Juraska, Geza Kovacs, Alison Lui, Ricardo Rei, Jason Riesa, Shruti Rijhwani, Parker Riley, Elizabeth Salesky, Firas Trabelsi, Stephanie Winkler, Biao Zhang, and Markus Freitag. 2025. WMT24++: Expanding the Language Coverage of WMT24 to 55 Languages & Dialects. *Preprint*, arXiv:2502.12404.

Dujian Ding, Ankur Mallick, Chi Wang, Robert Sim, Subhabrata Mukherjee, Victor Rühle, Laks V. S. Lakshmanan, and Ahmed Hassan Awadallah. 2024. Hybrid LLM: cost-efficient and quality-aware query routing. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*.

Shachar Don-Yehiya, Leshem Choshen, and Omri Abend. 2022. Prequel: Quality estimation of machine translation outputs in advance. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 11170–11183.

Zhichen Dong, Zhanhui Zhou, Zhixuan Liu, Chao Yang, and Chaochao Lu. 2025. Emergent response planning in llms. In *Forty-second International Conference on Machine Learning, ICML 2025, Vancouver, BC, Canada, July 13-19, 2025*.

António Farinhas, Nuno M Guerreiro, Sweta Agrawal, Ricardo Rei, and André FT Martins. 2025. Translate smart, not hard: Cascaded translation systems with quality-aware deferral. *arXiv preprint arXiv:2502.12701*.

Nuno M Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André FT Martins. 2024. xcomet: Transparent machine translation evaluation through fine-grained error detection. *Transactions of the Association for Computational Linguistics*, 12:979–995.

Neha Gupta, Harikrishna Narasimhan, Wittawat Jitkrittum, Ankit Singh Rawat, Aditya Krishna Menon, and Sanjiv Kumar. 2024. Language model cascades: Token-level uncertainty and beyond. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*.

Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan Awadalla. 2023. How good are gpt models at machine translation? a comprehensive evaluation. *arXiv preprint arXiv:2302.09210*.

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

Sunbowen Lee, Qingyu Yin, Chak Tou Leong, Jialiang Zhang, Yicheng Gong, Shiwen Ni, Min Yang, and Xiaoyu Shen. 2025. Probing the difficulty perception mechanism of large language models. *arXiv preprint arXiv:2510.05969*.

Yingfeng Luo, Ziqiang Xu, Yuxuan Ouyang, Murun Yang, Dingyang Lin, Kaiyan Chang, Tong Zheng, Bei Li, Peinan Feng, Quan Du, and 1 others. 2025a. Beyond english: Toward inclusive and scalable multilingual machine translation with llms. *arXiv preprint arXiv:2511.07003*.

Yingfeng Luo, Tong Zheng, Yongyu Mu, Bei Li, Qinghong Zhang, Yongqi Gao, Ziqiang Xu, Peinan Feng, Xiaoqian Liu, Tong Xiao, and JingBo Zhu. 2025b. Beyond decoder-only: Large language models can be good encoders for machine translation. In *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 9399–9431. Association for Computational Linguistics.

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

Lorenzo Proietti, Stefano Perrella, Vilém Zouhar, Roberto Navigli, and Tom Kocmi. 2025. Estimating machine translation difficulty. *Preprint*.

Xinyuan Wang, Yanchi Liu, Wei Cheng, Xujiang Zhao, Zhengzhang Chen, Wenchao Yu, Yanjie Fu, and Haifeng Chen. 2025. Mixllm: Dynamic routing in mixed large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2025 - Volume 1: Long Papers, Albuquerque, New Mexico, USA, April 29 - May 4, 2025*, pages 10912–10922.

Zhanglin Wu, Daimeng Wei, Xiaoyu Chen, Hengchao Shang, Jiaxin Guo, Zongyao Li, Yuanchang Luo, Jinlong Yang, Zhiqiang Rao, and Hao Yang. 2025. Combining the best of both worlds: A method for hybrid nmt and llm translation. In *Annual Meeting of the Association for Computational Linguistics*.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2024. A paradigm shift in machine translation: Boosting translation performance of large language models. In *ICLR*. OpenReview.net.

Wen Yang, Chong Li, Jiajun Zhang, and Chengqing Zong. 2023. Bigtrans: Augmenting large language models with multilingual translation capability over 100 languages. *CoRR*, abs/2305.18098.

Yubo Zhu, Dongrui Liu, Zecheng Lin, Wei Tong, Sheng Zhong, and Jing Shao. 2025. The llm already knows: Estimating llm-perceived question difficulty via hidden representations. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 1160–1176.

A Baseline Details

We compare against a diverse set of routing baselines, including random routing, heuristics, and learned routers. All routing methods output a scalar routing score s from the source sentence x (and direction d when applicable). Given a large-model budget p, we route the top-p (or bottom-p) fraction of requests by s to M_ℓ and the remainder to M_s.

- **Random.** Routes a uniformly random p fraction of requests to M_ℓ.
- **Heuristic routers.**
  - **Length:** Sentence length under tokenization (`spaCy` for En/Ru; `jieba` for Zh), with longer inputs prioritized.
  - **Rarity:** We compute unigram frequencies $f(w)$ from `wordfreq` and score each sentence by the average surprisal $-\log f(w)$ of its bottom 30% least-frequent tokens (higher $\Rightarrow$ higher routing priority).
  - **Entropy:** Entropy of M_s's next-token distribution at the first decoding step, used as a lightweight uncertainty proxy (higher $\Rightarrow$ higher routing priority).
- **Learned routers.** We consider learned routing with two system forms: *in-model* routers that leverage the small translator's representations, and *external* routers based on separately served encoder-only models. Each form is studied under two learning targets: predicting marginal gain (Δ) or predicting small-model quality (Q).
  - **In-model routers.**
    - **RouteLMT (Δ; ours).** The in-model gain router in §4 that predicts marginal gain.
    - **RouteLMT-Q.** Same architecture as RouteLMT, trained to predict the small model's quality.
  - **External encoder-only routers.**
    - **XLM-R-Δ.** An XLM-RoBERTa-Large model trained on the same supervision as RouteLMT to predict marginal gain from the source.
    - **XLM-R-Q.** An XLM-RoBERTa-Large model trained on the same supervision to predict the small model's quality from the source.
    - **sentinel-src-24/25** (Proietti et al., 2025). Publicly released XLM-RoBERTa-based source-only model trained for translation quality prediction.

Hyperparameter	Value
Train Type	LoRA
Learning Rate	1e-4
Number of Epochs	1
Global Batch Size	64
Max Length	1024
Warmup Ratio	0.05
LoRA Rank	8
LoRA Alpha	32
Target Modules	all-linear
Precision	bfloat16

Table 5: Hyperparameter configuration for the regression task.

Split	Source	Direction	# Size
Train	ComMT (General Translation)	En↔Zh En↔Ru	56,918 40,862
Eval	FLORES + BOUQuET + WMT24++	En↔Zh En↔Ru	3,368 3,368

Table 6: Training and evaluation datasets used in our experiments.

B Domain Generalization

In practice, a routing policy is exposed to a wide mix of user inputs, and the input distribution can across domains, so robustness to domain shift is important. To test robustness, we reuse the same routers from Table 2, trained on the ComMT General Translation split. We then evaluate them on medical (Table 8) and colloquial (Table 9) domain from the ComMT (Luo et al., 2025b) domain test sets without any additional adaptation. We evaluate under the same budgeted setting with p=0.3.

The overall ordering of methods is consistent with the main results in Table 2. On both domains, gain-based in-model routing remains the strongest practical approach. In particular, RouteLMT yields higher allocation benefit than surface heuristics and is competitive with or better than external source-only predictors. Ranking and selection metrics show the same pattern, indicating that the learned routing signal transfers under domain shift. These results indicate that routing signals derived from the small translator's prompt representations transfer well across domains.

Method	Type	Form	Target	Score	Route	Extra model
Random	–	–	–	uniform	top-p	✗
Length	Heuristic	–	–	length	top-p	✗
Rarity	Heuristic	–	–	surprisal	top-p	✗
Entropy	Heuristic	–	–	entropy	top-p	✗
XLM-R-Δ	Learned	External	Δ	$\hat{g}(x;d)$	top-p	✓
XLM-R-Q	Learned	External	Q	$\hat{q}_s(x;d)$	bottom-p	✓
sentinel-src-24/25 (Proietti et al., 2025)	Learned	External	Q	$\hat{q}_s(x;d)$	bottom-p	✓
RouteLMT (Δ; ours)	Learned	In-model	Δ	$\hat{g}(x;d)$	top-p	✗
RouteLMT-Q	Learned	In-model	Q	$\hat{q}_s(x;d)$	bottom-p	✗

Table 7: Baselines for budgeted hybrid routing. Route indicates whether we select the top-p or bottom-p fraction under budget p, depending on the score semantics. Δ denotes marginal-gain prediction and Q denotes small-model quality/difficulty prediction.

Method	Spearman ↑	HitRate@p ↑					MeanΔ@p ↑				
	Avg.	En→Zh	En→Ru	Zh→En	Ru→En	Avg.	En→Zh	En→Ru	Zh→En	Ru→En	Avg.
Gain Oracle	1.00	100.00	100.00	100.00	100.00	100.00	12.55	25.36	8.30	13.53	14.94
Quality Oracle	0.64	69.36	77.24	74.89	74.58	74.02	9.88	22.92	7.06	11.13	12.75
Random	0.00	30.00	30.00	30.00	30.00	30.00	3.00	9.27	1.95	3.04	4.32
Length	0.18	39.29	42.86	44.40	39.71	41.56	4.36	13.63	3.22	4.64	6.46
Rarity	0.14	32.20	42.86	38.58	40.92	38.64	3.30	12.99	2.68	5.07	6.01
Entropy	0.05	33.62	32.93	34.04	32.69	33.32	3.69	10.66	2.31	4.17	5.21
sentinel-src-24	0.26	42.55	49.64	47.38	49.88	47.36	4.99	15.21	3.78	6.43	7.60
sentinel-src-25	0.24	43.40	48.91	46.95	48.43	46.92	5.07	15.17	3.71	5.90	7.46
XLM-R-Δ	0.23	40.28	45.76	46.81	47.22	45.02	4.79	14.44	3.53	5.82	7.15
XLM-R-Q	0.18	39.72	43.34	45.67	41.89	42.66	4.81	13.86	3.41	5.09	6.79
RouteLMT-Q	0.30	46.52	53.03	48.94	**51.57**	51.04	**5.84**	16.43	3.99	**6.69**	8.24
RouteLMT	**0.30**	**47.66**	**54.72**	**53.03**	50.12	**51.39**	5.81	**16.78**	**4.39**	6.64	**8.41**

Table 8: Router performance on the *medical* domain across four translation directions. Spearman measures global ranking quality, while HitRate@p and MeanΔ@p are evaluated with a fixed budget p=0.3. **Bold** indicates the best result and underline the second best.

Method	Spearman ↑	HitRate@p ↑					MeanΔ@p ↑				
	Avg.	En→Zh	En→Ru	Zh→En	Ru→En	Avg.	En→Zh	En→Ru	Zh→En	Ru→En	Avg.
Gain Oracle	1.00	100.00	100.00	100.00	100.00	100.00	7.77	28.51	7.30	17.22	15.20
Quality Oracle	0.61	69.80	66.98	74.07	80.06	72.73	6.11	24.36	6.35	14.96	12.94
Random	0.00	30.00	30.00	30.00	30.00	30.00	1.49	11.84	1.38	4.74	4.86
Length	0.21	45.30	45.79	45.87	38.01	43.74	3.30	17.17	2.16	6.13	7.19
Rarity	0.14	33.90	36.45	42.17	41.12	38.41	1.99	14.15	2.85	6.88	6.47
Entropy	0.03	32.76	31.46	36.75	35.83	34.20	1.54	13.71	2.28	5.64	5.79
sentinel-src-24	0.26	44.73	48.60	52.71	51.09	49.28	3.12	17.72	2.76	8.72	8.08
sentinel-src-25	0.21	43.59	43.93	48.43	45.48	45.36	2.75	16.36	3.19	7.32	7.40
XLM-R-Δ	0.24	49.00	47.66	45.01	49.84	47.88	3.52	17.47	2.21	9.15	8.09
XLM-R-Q	0.19	47.86	46.11	42.45	44.86	45.32	3.44	16.66	1.84	7.62	7.39
RouteLMT-Q	0.29	**49.00**	48.91	51.85	**53.89**	50.91	3.48	18.17	3.86	**10.19**	**8.92**
RouteLMT	**0.30**	48.72	**50.47**	**52.99**	52.96	**51.28**	**3.68**	**18.50**	**3.90**	9.41	8.87

Table 9: Router performance on the *colloquial* domain across four translation directions. Spearman measures global ranking quality, while HitRate@p and MeanΔ@p are evaluated with a fixed budget p=0.3. **Bold** indicates the best result and underline the second best.

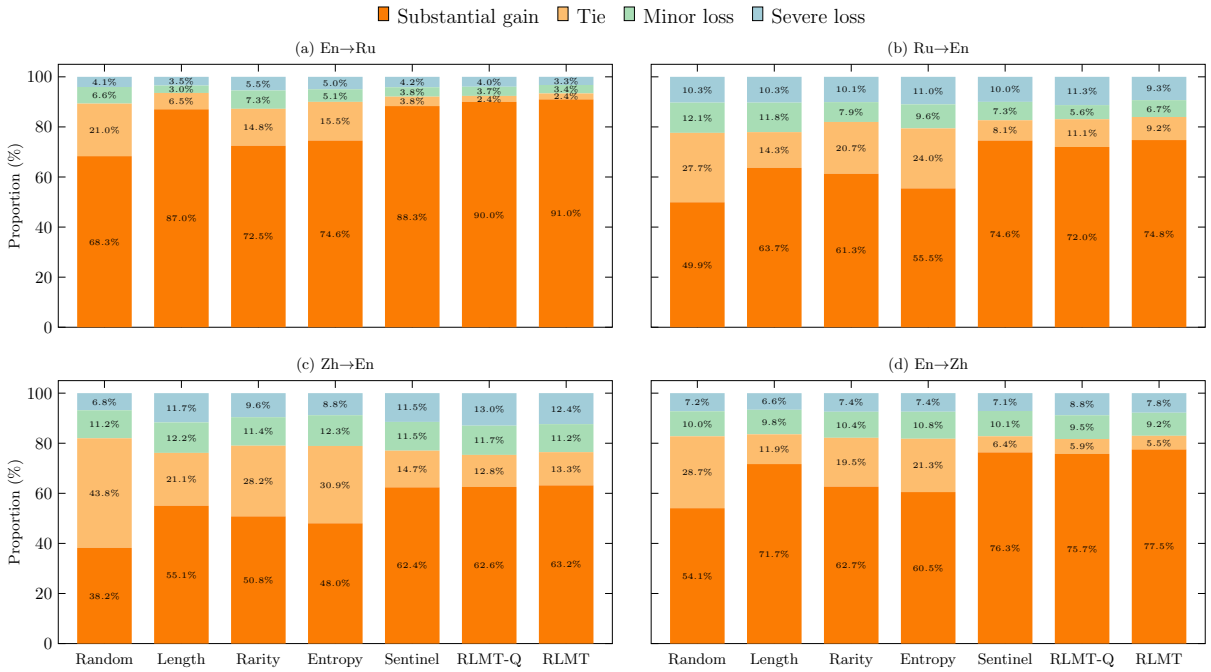

Figure 3: Gain-bucket distribution among routed-to-large model requests under budget p=0.3.