Confident or Seek Stronger: Exploring Uncertainty-Based Small LM Routing From Benchmarking to Generalization

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

Small language models (SLMs) are increasingly deployed on edge devices for personalized applications, offering efficient decoding latency and reduced energy consumption. However, these SLMs often generate inaccurate responses when handling complex queries. One promising solution is uncertainty-based SLM routing, offloading high-stakes queries to stronger large language models (LLMs) when resulting in low-confidence responses on SLM. This follows the principle of *If you lack confidence, seek stronger support* to enhance reliability. Relying on more powerful LLMs is yet effective but increases invocation costs. Therefore, striking a routing balance between efficiency and efficacy remains a critical challenge. Additionally, efficiently generalizing the routing strategy to new datasets remains under-explored. In this paper, we conduct a comprehensive investigation into ***benchmarking and generalization of uncertainty-driven routing strategies from SLMs to LLMs over 5000+ settings***. Our findings highlight: *First*, uncertainty-correctness alignment in different uncertainty quantification (UQ) methods significantly impacts routing performance. *Second*, uncertainty distributions depend more on both the specific SLM and the chosen UQ method, rather than on downstream data. Building on the insight, we propose a proxy routing data construction pipeline and open-source a hold-out set to enhance the generalization on predicting the routing curve for new downstream data. Experimental results indicate that proxy routing data effectively bootstraps routing performance without any new data. The source code is available at `https://anonymous.4open.science/r/quodlibeta`.

1 Introduction

Large language models (LLMs) deployment on edge devices has gained increasing attention in recent years, primarily due to their potential for low-latency, privacy-preserving inference. Given the computational and memory constraints of edge devices, small language models (SLMs) (e.g., Phi2-mini [35] or Llama3.2-3B [70] are designed for resource-efficient deployment, particularly on devices such as smartphones and wearable devices. Their overarching goal is to democratize the deployment of LMs, making it accessible and affordable to users across diverse settings and at any time [52, 86, 83]. However, these SLMs often lack the robustness and scalability of LLMs [8] (e.g., GPT-4o [2] and Llama-3.1-405B), especially when faced with diverse and complex input queries under the deployment on edge devices, which eventually degrade the overall performance. This limitation raises a critical need for exploring solutions to increase the response reliability of SLMs.

To mitigate this unreliability, a line of work proposes to partially offload challenging and complex queries from SLMs to LLMs [11, 59, 32, 66]. A hybrid system is then established to wisely route the queries from SLMs and seek more reliable and deterministic responses from stronger LLMs.

Table 1: Uncertainty quantification (UQ) methods evaluated in our benchmark. “Model Access” specifies whether a method views the LM’s weights/logits (white-box) or only its generated output (black-box). “Require Training?” indicates if additional training is needed. See Subsection 2.1 for taxonomy details and Subsection 3.1 for method descriptions.

Uncertainty Quantification (UQ) Methods	Taxonomy	Model Access	Require Training?
Average Token Prob [53]	Token/sequence probabilities	White-box	No
p(True) [39]	Token/sequence probabilities	White-box	No
Perplexity [21]	Token/sequence probabilities	White-box	No
Jaccard Degree [47]	Output consistency	Black-box	No
Verbalization-1s [76, 69]	Verbalized uncertainty	Black-box	No
Verbalization-2s [69]	Verbalized uncertainty	Black-box	No
Trained Probe [4, 39, 53]	Uncertainty probe	White-box	Yes
OOD Probe [39, 53]	Uncertainty probe	White-box	Yes

Although LLMs can exhibit superior performance, they incur high maintenance and inference costs given the large scale of model size and their infrastructure (i.e., a single NVIDIA A100 GPU can cost approximately $2,000 per month for deployment). Inaccurate routing by SLMs increases the volume of queries forwarded to LLMs, necessitating greater bandwidth allocation for maintaining the service of LLMs. As a result, operational costs and budgetary requirements rise accordingly, especially when continuous deployment is required. Hence, developing an effective routing strategy is crucial for fully deploying SLMs [59, 66, 11], as it both enhances response reliability and reduces the costs associated with services and data transmission.

Leveraging SLMs’ self-uncertainty estimation emerges as a robust strategy for enhancing routing effectiveness [11, 16]. By relying on the self-assessed uncertainty, the system can better decide whether to handle a query locally or delegate it to a larger model without the aid of extra routers, ensuring that only queries deemed unreliable by the SLMs are routed to LLMs. As a result, the uncertainty-based routing approach not only generalizes well to new datasets, as only self-assessed information from SLM is needed, but it also reduces the high operational costs associated with accurately running LLMs. To this end, we aim to explore two open and nontrivial research questions for uncertainty-based SLM routing:

1) What is the best practice of uncertainty estimation for query routing from SLMs to LLMs? In this research question, we benchmark the uncertainty-correctness alignment of each uncertainty quantification (UQ) method under its impact on SLM routing. A good alignment is a key factor for successful routing decisions, as any misalignment can cause unnecessary offloading with extra cost. However, SLMs may struggle to provide reliable uncertainty estimates [33, 15, 73], making them less effective as indicators for query routing. Thus, we benchmark the alignment between uncertainty and correctness, paving the insights for establishing more effective routing strategies[1].

2) What is the best practice to initially establish an effective routing strategy when generalizing to new datasets? In this research question, we explore how to generalize routing strategies to new datasets. Existing approaches [59, 32] rely on sufficient new downstream data to make routing decisions for optimal performance-cost trade-offs, but this process is time-consuming and labor-intensive. Broadly speaking, collecting and analyzing full downstream datasets under varying SLM configurations can be prohibitively costly, delaying implementation, which is not practical in real-world scenarios. This delay is particularly problematic in high-stakes scenarios, such as medical wearable devices, where reliability is critical, and inaccuracies are unacceptable even in early deployment stages. Based on our findings, we provide a data construction pipeline to predict the routing curves in new downstream scenarios without any new downstream data. A generated proxy routing dataset as a data-agnostic hold-out set enables the estimation of effective routing decisions via the predicted routing curves. We further benchmark the benefits of this proxy routing dataset, demonstrating its generalization ability in predicting the routing curve to new datasets.

This work offers an accessible and reproducible pipeline for uncertainty-based routing from benchmarking to generalization. Our main contributions are summarized as follows:

[1]For the convenience of writing, we interchangeably use uncertainty and confidence, where low uncertainty refers to high confidence.

- **Comprehensive benchmarking and detailed analysis:** This benchmark evaluates 8 UQ methods across 14 datasets to examine the alignment between uncertainty and correctness in routing tasks. We incorporate 8 SLMs and 2 LLMs to emulate real-world deployment scenarios. We then delve into key observations from the extensive results and conclude the insights for developing uncertainty-based SLM routing.
- **Proxy routing data for generalizing routing to new data:** Building on our benchmarking pipeline, we introduce a proxy routing data construction pipeline designed to generalize the routing curve prediction in new downstream scenarios. Empirical results show that this proxy routing data generalizes effectively the routing prediction to new datasets *without relying on any new downstream data*.

2 Reviewing Different Schools of Uncertainty Quantification and LLM Routing

2.1 Uncertainty Quantification for LMs

Uncertainty quantification methods estimate a model's confidence in its predictions [31]. For traditional classification and regression, uncertainty estimation is well-established [23]. However, for LLMs generating free-form responses to complex queries, estimating uncertainty is more challenging because the output space can grow exponentially with vocabulary size, and each sequence spans multiple tokens [20]. Existing uncertainty quantification approaches for LLMs can be grouped into the following four categories.

Via verbalizing uncertainty. This line of work prompts language models to report linguistic confidence [53, 56]. To enable LMs to verbalize confidence, researchers have proposed fine-tuning them to express uncertainty [46] or teaching them to verbalize confidence through in-context learning [17]. Verbalized confidence can take the form of linguistic expressions of uncertainty or numerical scores [24]. Multiple studies find that LLMs tend to be overconfident when reporting confidence [76, 69]. To mitigate this overconfidence, prompting strategies such as multi-step elicitation, top-k, and Chain-of-Thought [72] have been explored [69]. Sampling multiple response-confidence pairs and designing more effective aggregation strategies can also help mitigate overconfidence [76]. Moreover, [69] reports that verbalized confidence is typically better calibrated than the model's conditional probabilities.

Via analyzing token/sequence probabilities. This line of research derives confidence scores from model logits for output tokens [24, 33, 38]. The confidence of a generated sequence is computed by aggregating the log-probabilities of its tokens. Common aggregation strategies include arithmetic average, minimum, perplexity, and average entropy [20, 21, 71]. Because not all tokens in a sequence equally reflect semantic content, SAR reweights token likelihoods to emphasize more meaningful tokens [18]. However, different surface realizations of the same claim can yield different probabilities, implying that the calculated confidence reflects how a claim is articulated rather than the claim itself [53]. To combine LM self-assessment with token probabilities, $p(\text{True})$ is proposed: the model is asked whether its generated response is correct, and the probabilities of True/False tokens serve as the confidence score [39, 69].

Via gauging output consistency. This line of research (e.g., SelfCheckGPT [54]) assumes that high-confidence LLMs produce consistent outputs [53]. A typical approach samples m responses for a given input query, measures inter-response similarity, and calculates a confidence score from meaning diversity [20]. Common ways to measure pairwise similarity include Natural Language Inference (NLI) and Jaccard similarity [24]. Consistency is then assessed by analyzing the similarity matrix, for instance, by counting semantic sets, summing eigenvalues of the graph Laplacian or computing eccentricity [47]. Because different sentences can express the same meaning, semantic entropy [40] first clusters responses by semantic equivalence before measuring consistency.

Via training uncertainty probes. This approach trains classifiers to predict whether an LLM will arrive at the correct answer for a particular query, using predicted probabilities as confidence scores [24]. Training data is often obtained by sampling multiple answers per question at a fixed temperature and labeling each for correctness [39]. A probe (commonly a multi-layer perceptron) then takes hidden states as inputs to predict correctness [4, 42]. Because in-domain training data is not always available, Contrast-Consistent Search trains probes unsupervisedly by maximizing

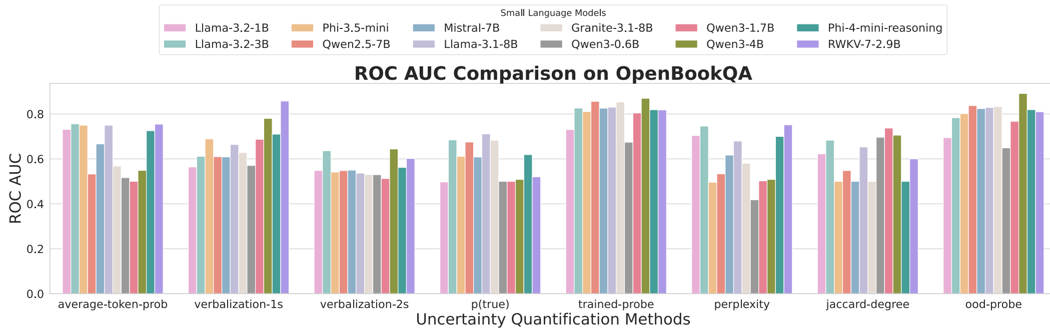


Figure 1: The ROC AUC scores measure the alignment between confidence and correctness across different SLMs and uncertainty quantification methods on OpenBookQA. A higher ROC AUC indicates a stronger alignment.

representation distances between contradictory answers on Yes/No questions [7]. Furthermore, whether probes trained on out-of-distribution data remain effective is still under debate [39, 53, 40].

2.2 LLM Routing

In query-routing scenarios, recent approaches train additional classifiers to direct queries to different SLMs or LLMs based on historical performance metrics and user feedback data [16, 59, 66, 37, 84]. For instance, RouterBench [32] collects inference outputs from selected LLMs to aid in the development of routing classifiers. However, these methods face significant challenges when encountering new downstream tasks, as such data falls outside the distribution of the existing training data. This limitation makes them less practical for real-world scenarios, such as on personal edge device deployment, where adaptability to unseen conditions is crucial. Our work focuses on how to establish routing systems between SLMs and LLMs and generalize to new downstream tasks. In this manner, uncertainty-based routing is an appropriate solution to overcome these challenges, as uncertainty is directly extracted from SLMs themselves. Furthermore, we propose a proxy routing data construction pipeline to initialize a routing system that generalizes to unseen datasets.

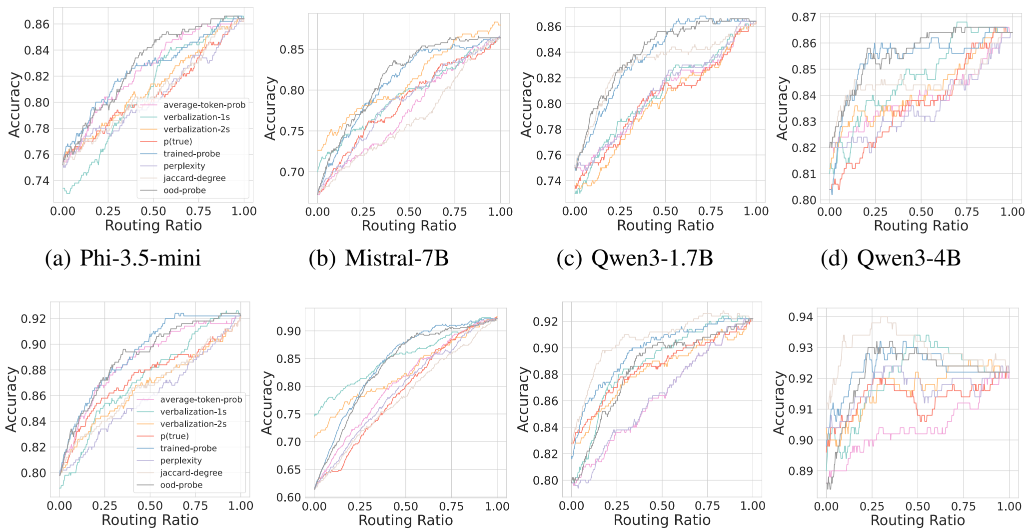
3 Benchmarking Uncertainty-based SLM Routing

In this section, we systematically evaluate 12 SLMs and 4 LLMs on 15 datasets using 8 UQ methods (see Table 1) for uncertainty-based SLM routing. This section details the datasets, models, and UQ methods, followed by several key findings and practical considerations. All experiments are conducted on four 80GB NVIDIA A100 GPUs.

3.1 Benchmark Coverage and Setup

Language Models. We evaluate 12 open-source SLMs, organized into three categories: non-reasoning LMs, reasoning LMs, and a recurrent neural network (RNN) model. The non-reasoning models are Llama-3.2-1B-Instruct [55], Llama-3.2-3B-Instruct [55], Phi-3.5-mini-instruct [1], Mistral-7B-Instruct-v0.3 [36], Qwen2.5-7B-Instruct [78], Llama-3.1-8B-Instruct [19], and Granite-3.1-8B-Instruct [26]. The reasoning models are Qwen3-0.6B, Qwen3-1.7B, Qwen3-4B, and Phi-4-mini-reasoning [77]. The RNN model is RWKV-7-2.9B [61]. These SLMs come from Alibaba (four models), Meta (three), Microsoft, Mistral AI, IBM, and LF AI & Data. Except for RWKV-7-2.9B, all adopt decoder-only Transformer architectures and are available on Hugging Face. We also include four LLMs: three open-source models—Llama-3.1-70B-Instruct [19], Qwen3-32B, and DeepSeek-R1 [29]—and one proprietary API model, GPT-4.1 mini [34]. Qwen3-32B and DeepSeek-R1 are reasoning LLMs, whereas Llama-3.1-70B and GPT-4.1 mini are non-reasoning.

Datasets. Experiments span 15 datasets from four domains: *(1) Mathematical Reasoning* (AQuA [48], GSM8K [13], MultiArith [63], SVAMP [60], MATH-500 [43]), *(2) Commonsense Reasoning* (CommonsenseQA [67], HellaSwag [80], OpenBookQA [57], PIQA [6], TruthfulQA [45], WinoGrande [64], BoolQ [12], Social IQa [65]), *(3) Conversational and Contextual Understanding*



(a) Phi-3.5-mini (b) Mistral-7B (c) Qwen3-1.7B (d) Qwen3-4B

(e) Phi-3.5-mini (f) Mistral-7B (g) Qwen3-1.7B (h) Qwen3-4B

Figure 2: Overall accuracy vs. routing ratio with different UQ methods and SLMs. (a)-(d) show the results of routing to DeepSeek-R1 on the CommonsenseQA dataset; and (e)-(h) demonstrate the results of routing to GPT-4.1 mini on the OpenBookQA dataset.

(CoQA [62]), and *(4) Problem Solving* (MMLU [30]). These cover free-form, multiple-choice, and True/False question answering and are available via Hugging Face. Table 2 in Appendix A provides further details.

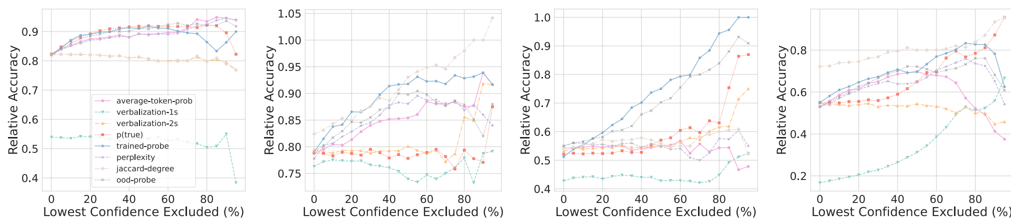
UQ Methods and Hyperparameters. We evaluate 8 approaches from the four categories in Section 2.1. *(1) Average token probability* uses the probability of the chosen option token (e.g., "A") for multiple-choice tasks or the mean probability of all generated tokens for free-form tasks. *(2) Perplexity* is computed for a sequence of N output tokens $\{y_i\}_{i=1}^N$ with probabilities $\{p(y_i)\}_{i=1}^N$ as $\exp\big(\frac{1}{N}\sum_{i=1}^N \ln p(y_i)\big)$, and its reciprocal serves as the confidence score. *(3) p(True)* is a method where the LM first outputs an answer, then evaluates the generated response using only "True" or "False." The probabilities for these two tokens are normalized to sum to 1, and the probability of "True" is used as confidence. *(4) Verbalized confidence in a single response* (denoted as verbalization-1s) prompts the model to output both the answer and numeric confidence in one step. *(5) Verbalized confidence in the second round* (denoted as verbalization-2s) obtains the confidence in a separate, follow-up query after the model has provided an answer. *(6) The degree matrix* (denoted as jaccard-degree) generates $m = 5$ samples (temperature 1.0) for one query, computes pairwise Jaccard similarities, and sets confidence to $\mathrm{trace}(mI - D)/m^2$, where D is the degree matrix. *(7) Trained probe* is a four-layer MLP with LeakyReLU activations, trained on a fixed subsample of the in-domain training set for each dataset, taking as input the hidden states from the eighth-to-last transformer layer. We train for 20 epochs (learning rate 5×10^{-4}). *(8) Trained probe on out-of-distribution data* (denoted as ood-probe) is identical in architecture but trained on all other datasets. e.g., if AQuA is evaluated, the ood-probe is trained on the remaining 14 datasets (20 epochs, learning rate 1×10^{-4}).

For verbalization-based methods, we discard queries when the model does not follow instructions to produce a confidence score. For free-form question answering, we use GPT-4.1 mini to evaluate whether a response is essentially equivalent to the ground truth answer [85].

3.2 Report Observations

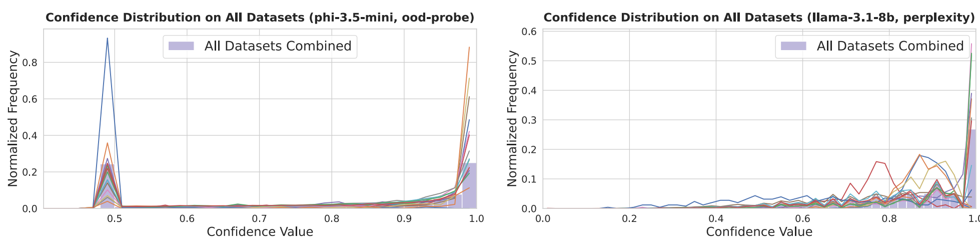
In this section, we present our benchmarking results analyzing the impact of uncertainty-correctness alignment on routing tasks. More observations and experimental results on proxy routing and routing can be found in Appendix C.1.

Observation ❶: Uncertainty estimation in SLMs may exhibit misalignment with prediction correctness. From the theoretical perspective, well-calibrated uncertainty scores do not necessarily



(a) Granite-3.1-8B (b) Qwen3-0.6B (c) Llama-3.2-1B (d) RWKV-7-2.9B

Figure 3: Relative accuracy of SLMs vs. LLMs on top-$k\%$ confident queries. "Relative accuracy" is the ratio of SLM accuracy to LLM accuracy. The x-axis "Lowest Conf. Excluded" shows the percentage of low-confidence queries removed; for example, 80 means 80% of queries with the lowest confidence are excluded, leaving the top 20%. (a) and (b) compare SLMs to Llama-3.1-70B on GSM8K, while (c) and (d) compare SLMs to Qwen3-32B on BoolQ.



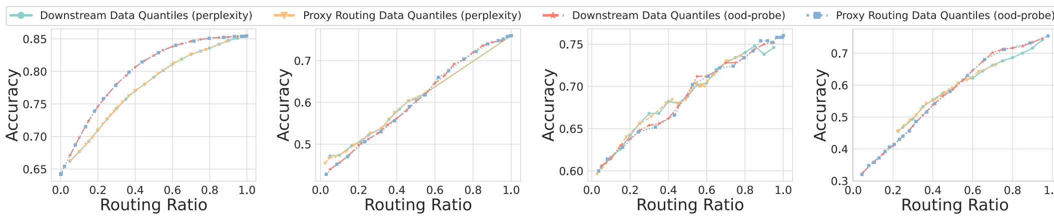
(a) Phi-3.5-mini (b) Llama-3.1-8B

Figure 4: Confidence distributions across 15 datasets. The histogram depicts the aggregated distribution from all datasets, while each curve represents a single dataset. (a) Confidence of Phi-3.5-mini by OOD Probe; (b) Confidence of Llama-3.1-8B by Perplexity.

imply a strong correlation with the correctness of the predictions [33, 11]. The predictions of models might be perfectly calibrated yet still display relatively low accuracy (i.e., confidently provide wrong answers). This phenomenon is also evident in our benchmark results (illustrated in Figure 1). We compute AUC scores to quantify the correlation between extracted uncertainty and prediction correctness, treating correctness as a binary ground truth and using confidence values as the ranking metric. The results show that not all UQ methods effectively exhibit a strong alignment between confidence and prediction correctness. Moreover, from Figure 1 and Figure 8, we can observe that the alignment may vary across datasets for the same SLM and UQ method. For instance, Perplexity [21] demonstrates strong alignment for Phi-3.5-mini on the MultiArith dataset but fails on the OpenBookQA dataset. On the other hand, OOD Probe, Trained Probe, and Perplexity obtain consistently decent alignment compared to other UQ methods across different SLMs and domains of datasets. Conversely, we notice that verbalization-based methods, namely verbalization-1s [69, 53], and verbalization-2s [69], consistently withhold low alignment between uncertainty and prediction correctness. More experimental results can be found in Appendix C.1.

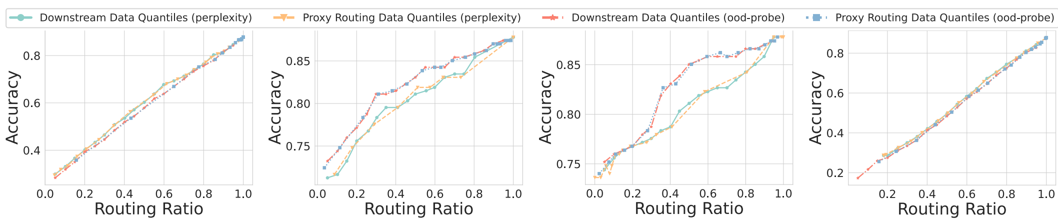
Observation ❷: Verbalization-based UQ methods struggle to extract uncertainty in SLMs for query routing. We find that verbalization methods like verbalization-2s [69] obtain poor alignment between confidence and prediction correctness, and this misalignment can lead to inferior routing performance in SLMs, where the conclusion can be found in Figure 2. Recent advancements [75, 79] also show that uncertainty scores derived from verbalization may exhibit good reflection on models' intrinsic uncertainty of prediction across multiple models and datasets. This discrepancy poses a significant challenge for establishing effective routing performance since queries that are actually correct may be unnecessarily routed from SLMs to LLMs, thereby increasing the overall cost of deploying routing systems.

Observation ❸: A good routing standard highly depends on UQ methods with good uncertainty-correctness alignment. A notable phenomenon occurs when UQ methods, such as Trained Probe [53], exhibit strong alignment, leading to significant improvements in routing performance. This is because



(a) Mistral-7B (b) Qwen3-0.6B (c) Phi-4-mini-reasoning (d) RWKV-7-2.9B

Figure 5: Routing results from four SLMs to Llama-3.1-70B on HellaSwag, with the remaining 14 other datasets constituting the proxy routing data.



(a) Mistral-7B (b) Qwen3-0.6B (c) Phi-4-mini-reasoning (d) RWKV-7-2.9B

Figure 6: Routing results from four SLMs to DeepSeek-R1 on AQuA (mathematical reasoning), with eight commonsense-reasoning datasets and one conversational & contextual understanding dataset constituting the proxy routing data, demonstrate the strong generalization capability of the proxy routing data in predicting the routing curve.

the extracted uncertainty scores from these UQ methods more effectively indicate whether SLMs produce correct predictions. Among all UQ methods evaluated for routing tasks, we find that Trained Probe [53], OOD Probe [39, 53], and Perplexity [20] consistently rank as the top three methods for SLM routing. Therefore, a comprehensive analysis of UQ methods before deploying a routing system in SLMs is highly recommended to ensure efficient query routing.

Observation ❹: SLMs can match LLM performance on high-confidence queries. Although SLMs generally underperform LLMs, we find that for queries where SLMs exhibit high confidence, their accuracy approaches that of LLMs. To illustrate, we progressively remove queries starting from those with the lowest SLM confidence and compute the ratio of SLM to LLM accuracy on the remaining top-$k\%$ queries (Figure 3). As more low-confidence queries are excluded, SLMs achieve comparable performance to LLMs. For instance, on GSM8K, Qwen3-0.6B achieves performance nearly equal to Llama-3.1-70B on the top 20% highest-confidence queries. Moreover, the effectiveness of this selection depends on the uncertainty quantification (UQ) method: approaches with stronger alignment (e.g., Trained Probe [53]) yield higher relative accuracy than weaker ones (e.g., verbalization-2s) across all query exclusion rates. Additional results appear in Appendix C.2.

4 Generalizable SLM Routing for New Downstream Scenarios

In this section, we first describe the pipeline for constructing proxy routing data with experimental details. We then investigate how well the proxy routing data can predict the routing curve for new downstream scenarios without accessing the new datasets. Finally, we discuss our results and offer several insights into the proxy routing data for establishing routing in early-stage deployments.

4.1 Proxy Routing Data Construction Pipeline

We aim to evaluate the effectiveness of proxy routing data in generalizing the routing curve predictions to new downstream scenarios, without relying on additional downstream data. Specifically, the proxy routing data serves as a data-agnostic hold-out set tailored to a particular SLM, which can generalize its routing standards across various new downstream datasets. By leveraging this proxy routing data, we establish a generalizable routing framework for the routing deployments in the new scenario.

Algorithm 1 Proxy Routing Data Construction Pipeline

input A collection of datasets $\mathbb{D} = \{\mathcal{D}_i\}_{i=1}^N$ with N domains
output A set of proxy routing data $\boldsymbol{X}$

Collect diverse domain of dataset $\mathcal{D}_i$ to form $\mathbb{D} = \{\mathcal{D}_i\}_{i=1}^N$
Generate uncertainty distributions $\{\mathcal{F}_\mathbb{D}\}_{i=1}^M$ of $\mathbb{D}$ with selected UQ methods
Sample $\boldsymbol{X} = \{x_j \mid x_j \in \boldsymbol{X}_i \sim \{\mathcal{F}_\mathbb{D}\}_{i=1}^M \ \forall i, j\}$ from i-th bin in $\{\mathcal{F}_\mathbb{D}\}_{i=1}^M$

This approach simplifies deployment of routing systems by eliminating the need for dataset-specific routing analysis and demonstrates that proxy routing data can generalize across diverse datasets.

The overall construction pipeline is detailed as follows. Let $\mathbb{D} = \{\mathcal{D}_i\}_{i=1}^N$ be a diverse collection of datasets, where N denotes the number of distinct domain types included in the collection. We select a diverse collection of datasets $\mathbb{D}$ with various domains, such as commonsense reasoning, mathematics, and more, where we follow the settings in [50]. And then, we process every data instance in $\mathbb{D}$ through selected UQ methods to capture their corresponding uncertainty distributions $\{\mathcal{F}_\mathbb{D}\}_{i=1}^M$ with M bins, where $M \in \mathbb{Z}^+$ is an arbitrary number. These distributions serve as the sampling foundation of each data instance in forming proxy routing data. Finally, data instances obtained in the set of proxy routing data $\boldsymbol{X}$ are weighted-sampled from each bin of $\{\mathcal{F}_\mathbb{D}\}_{i=1}^M$ such that $\boldsymbol{X} = \{x_j \mid x_j \in \boldsymbol{X}_i \sim \{\mathcal{F}_\mathbb{D}\}_{i=1}^M \ \forall i, j\}$. This ensures similar distribution in proxy routing data across various uncertainty levels presented in $\{\mathcal{F}_\mathbb{D}\}_{i=1}^M$. The resulting collection of these sampled data instances forms the final proxy routing dataset. The detailed pipeline of constructing the proxy routing dataset is outlined in Algorithm 1.

4.2 Proxy Routing Data Setups

Benchmark Settings. We evaluate the constructed proxy routing data on 15 SLMs and 4 LLMs across 15 datasets. Based on the observations and results from the previous benchmark section, we select 2 UQ methods that demonstrate the strongest alignment between predicted uncertainty and actual correctness: "OOD Probe" [39, 53] and "Perplexity" [20] method. We consider the routing performance evaluated on the entire new dataset as the ground truth. To simulate new dataset scenarios, we introduce two evaluation settings: (1) fully out-of-domain and (2) partially in-domain. First, for the out-of-domain setting, we evaluate a target dataset using proxy routing data derived from source datasets with no domain overlap. Second, in the partially in-domain setting, we designate one dataset as the target and construct its proxy routing data using the remaining 14 datasets, where the domain of the dataset may partially overlap. The target dataset's generalization performance is then evaluated using this proxy routing set, which does not contain any information from the target dataset. All reported results represent the average across three individual experimental runs.

Data Construction Settings. The proxy routing data is weighted-sampled from each bin of the proxy routing data distributions, with the number of bins set to 30. We sample 10% of the instances from each bin to form the final proxy routing data. The temperature is fixed at 0 with a fixed random seed of 50 to ensure reproducibility.

4.3 Routing Curve Prediction with Proxy Routing Data

We provide several key insights into the generalization ability of proxy routing data as follows.

Insights ❶: The extracted confidence distribution is predominantly determined by the chosen SLM and uncertainty quantification (UQ) method, with minimal dependence on the downstream dataset. As illustrated in Figure 4, confidence scores aggregated from 15 different tasks exhibit a nearly identical shape regardless of the specific dataset. Instead, they vary notably with different SLMs and UQ methods. This finding suggests that the confidence distribution is largely data-agnostic, enabling the construction of proxy routing data that generalizes to new tasks without any new datasets.

Insights ❷: Proxy routing data helps SLM routing to predict an accurate routing curve without any new data, allowing routing strategies to be initialized on SLMs without accessing new datasets. Building on our findings about uncertainty distributions, we sampled a data subset to create a final proxy routing dataset using the pipeline described in Section 4.1. We then utilized this proxy routing dataset to predict all thresholds for different routing ratios in new downstream scenarios.

The experimental results (see Figure 5 and Figure 6) show that the routing curves from the proxy routing data closely match those from the entire new downstream dataset in both evaluation settings, indicating that the proxy routing data provides strong capability for establishing routing strategies on unseen downstream datasets. An identical phenomenon is observed across multiple UQ methods and different SLMs, highlighting the potential of proxy routing data to initiate the routing process for any new dataset, independent of the UQ method or SLM used. More results are in Appendix C.3.

5 Challenges and Opportunities

❶ **How to cash-in routing efficiency on new edge devices?** Based on the benchmark results, proxy routing data provides a robust foundation for establishing routing policies on new edge devices without accessing prior knowledge at the early stage of deployment. This enables the routing policies with strong generalization to new dataset scenarios and enhances the efficiency across diverse deployments for personal edge devices. While proxy routing data holds a good performance in the early deployment stage, an important direction to explore is how to effectively leverage additional private on-device data to strengthen the quality of proxy routing data, aiming to continuously enhance the deployment of personalized routing strategies. With the aid of proxy routing data, less private data is required, but striking a balance between privacy and performance remains an open challenge.

❷ **How to effectively strike a balance between LLM routing efficiency and utility?** We empirically observe that by leveraging UQ methods with strong uncertainty-utility alignment (e.g., Perplexity and OOD Probe methods), routing thresholds can effectively be determined with the sweet points of efficiency and utility. However, achieving such sweet spots can be challenging due to the variability in downstream datasets and the sensitivity of UQ methods to LLM-specific characteristics. Additionally, discrepancies across different device types, such as variations between iOS and Android systems , further complicating the process, requiring tailored strategies and analytics to account for platform-specific constraints and capabilities. Based on these factors, providing a fair apple-to-apple comparison regarding routing performance is inherently challenging. Researchers should be mindful of these complexities and focus on developing methods that are not only efficient but also capable of handling long-context scenarios effectively.

❸ **How is the performance when conducting compression (e.g., pruning, quantization) on the on-device model?** As with the on-device models discussed in the above sections, we directly adopt a pre-trained small model without any modifications. Alternatively, on-device models can also be generated by compressing larger models. Specifically, numerous works have explored methods for compressing LLMs into smaller sizes using techniques such as pruning [22] and quantization [74, 44]. The advantage of employing compression methods is that the smaller models compressed from larger ones tend to retain similar distributions of the output, thereby mitigating the issue of distribution shift.

❹ **Uncertainty-aware routing in on-device multimodal language models.** While LLMs typically operate with a single modality for both input and output, a promising research direction involves exploring uncertainty-aware routing in multimodal language models (MLLMs). For instance, in vision-language models (VLMs) such as LLaVa [49] and InternVL [9], the inputs include both images/videos and text. By incorporating visual modalities, the properties of vision tokens significantly influence the output. As a result, the uncertainty in the generated text differs from that of language-only models. Benchmarking and generalizing uncertainty-aware routing for on-device MLLMs is a valuable direction for the research community.

6 Conclusion

This paper investigates the routing accuracy of SLMs in estimating their uncertainty and establishing best practices for initiating effective routing strategies. Through comprehensive benchmarking of 15 SLMs, 4 LLMs, 8 UQ methods, and 15 datasets across 5000+ settings, we found that the alignment between uncertainty and correctness significantly impacts routing performance. Additionally, our experiments show that uncertainty distributions depend primarily on the specific SLM and UQ method rather than the downstream data. Building on the insights, we introduced a proxy routing data construction pipeline and a hold-out dataset to generalize routing strategies without prior knowledge of new downstream data. The results confirm that the proxy routing data effectively bootstraps routing, indicating its strong potential for benefiting in resource-efficient SLM deployment.

References

[1] Marah Abdin, Jyoti Aneja, Hany Awadalla, Ahmed Awadallah, Ammar Ahmad Awan, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Jianmin Bao, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

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

[3] Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Leandro von Werra, and Thomas Wolf. Smollm - blazingly fast and remarkably powerful, 2024.

[4] Amos Azaria and Tom Mitchell. The internal state of an llm knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, 2023.

[5] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023.

[6] Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439, 2020.

[7] Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. Discovering latent knowledge in language models without supervision. *arXiv preprint arXiv:2212.03827*, 2022.

[8] Lihu Chen and Gaël Varoquaux. What is the role of small models in the llm era: A survey. *arXiv preprint arXiv:2409.06857*, 2024.

[9] Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 24185–24198, 2024.

[10] Xiangxiang Chu, Limeng Qiao, Xinyu Zhang, Shuang Xu, Fei Wei, Yang Yang, Xiaofei Sun, Yiming Hu, Xinyang Lin, Bo Zhang, et al. Mobilevlm v2: Faster and stronger baseline for vision language model. *arXiv preprint arXiv:2402.03766*, 2024.

[11] Yu-Neng Chuang, Helen Zhou, Prathusha Kameswara Sarma, Parikshit Gopalan, John Boccio, Sara Bolouki, and Xia Hu. Learning to route with confidence tokens. *arXiv preprint arXiv:2410.13284*, 2024.

[12] Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, 2019.

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

[14] Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. *arXiv preprint arXiv:2405.21060*, 2024.

[15] Gianluca Detommaso, Martin Bertran, Riccardo Fogliato, and Aaron Roth. Multicalibration for confidence scoring in llms. *arXiv preprint arXiv:2404.04689*, 2024.

[16] Dujian Ding, Ankur Mallick, Chi Wang, Robert Sim, Subhabrata Mukherjee, Victor Rühle, Laks VS Lakshmanan, and Ahmed Hassan Awadallah. Hybrid llm: Cost-efficient and quality-aware query routing. In *The Twelfth International Conference on Learning Representations*, 2024.

[17] Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Jingyuan Ma, Rui Li, Heming Xia, Jingjing Xu, Zhiyong Wu, Baobao Chang, et al. A survey on in-context learning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1107–1128, 2024.

[18] Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5050–5063, 2024.

[19] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[20] Ekaterina Fadeeva, Roman Vashurin, Akim Tsvigun, Artem Vazhentsev, Sergey Petrakov, Kirill Fedyanin, Daniil Vasilev, Elizaveta Goncharova, Alexander Panchenko, Maxim Panov, et al. Lm-polygraph: Uncertainty estimation for language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 446–461, 2023.

[21] Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:539–555, 2020.

[22] Elias Frantar and Dan Alistarh. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *International Conference on Machine Learning*, pages 10323–10337. PMLR, 2023.

[23] Yarin Gal et al. Uncertainty in deep learning. 2016.

[24] Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. A survey of confidence estimation and calibration in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6577–6595, 2024.

[25] Xinyang Geng and Hao Liu. Openllama: An open reproduction of llama, May 2023.

[26] IBM Granite Team. Granite 3.0 language models, 2024.

[27] RLHF Griffin and Gemma Teams. Recurrentgemma: Moving past transformers for efficient open language models, 2024.

[28] Dirk Groeneveld, Iz Beltagy, Pete Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Harsh Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew E. Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, Will Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah A. Smith, and Hannaneh Hajishirzi. Olmo: Accelerating the science of language models. *Preprint*, 2024.

[29] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

[30] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

[31] Bairu Hou, Yujian Liu, Kaizhi Qian, Jacob Andreas, Shiyu Chang, and Yang Zhang. Decomposing uncertainty for large language models through input clarification ensembling. In *Forty-first International Conference on Machine Learning*, 2024.

[32] 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.

[33] Yuheng Huang, Jiayang Song, Zhijie Wang, Shengming Zhao, Huaming Chen, Felix Juefei-Xu, and Lei Ma. Look before you leap: An exploratory study of uncertainty measurement for large language models. *arXiv preprint arXiv:2307.10236*, 2023.

[34] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.

[35] Mojan Javaheripi and Sébastien Bubeck. Phi-2: The surprising power of small language models, December 2023.

[36] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b. *arXiv*, 2023.

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

[38] Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 2021.

[39] Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

[40] Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*, 2023.

[41] Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. Bloom: A 176b-parameter open-access multilingual language model. 2023.

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

[43] Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *The Twelfth International Conference on Learning Representations*, 2023.

[44] Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. *Proceedings of Machine Learning and Systems*, 6:87–100, 2024.

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

[46] Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*, 2022.

[47] Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. Generating with confidence: Uncertainty quantification for black-box large language models. *arXiv preprint arXiv:2305.19187*, 2023.

[48] Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. Program induction by rationale generation: Learning to solve and explain algebraic word problems. *ACL*, 2017.

[49] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36, 2024.

[50] Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation. *arXiv preprint arXiv:2402.09353*, 2024.

[51] Zechun Liu, Changsheng Zhao, Forrest Iandola, Chen Lai, Yuandong Tian, Igor Fedorov, Yunyang Xiong, Ernie Chang, Yangyang Shi, Raghuraman Krishnamoorthi, et al. Mobilellm: Optimizing sub-billion parameter language models for on-device use cases. *arXiv preprint arXiv:2402.14905*, 2024.

[52] Zhenyan Lu, Xiang Li, Dongqi Cai, Rongjie Yi, Fangming Liu, Xiwen Zhang, Nicholas D Lane, and Mengwei Xu. Small language models: Survey, measurements, and insights. *arXiv preprint arXiv:2409.15790*, 2024.

[53] Matéo Mahaut, Laura Aina, Paula Czarnowska, Momchil Hardalov, Thomas Müller, and Lluís Màrquez. Factual confidence of llms: on reliability and robustness of current estimators. *arXiv preprint arXiv:2406.13415*, 2024.

[54] Potsawee Manakul, Adian Liusie, and Mark Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, 2023.

[55] AI Meta. Llama 3.2: Revolutionizing edge ai and vision with open, customizable models. *Meta AI Blog. Retrieved December*, 20:2024, 2024.

[56] Sabrina J Mielke, Arthur Szlam, Emily Dinan, and Y-Lan Boureau. Reducing conversational agents' overconfidence through linguistic calibration. *Transactions of the Association for Computational Linguistics*, 10:857–872, 2022.

[57] Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*, 2018.

[58] Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Pete Walsh, Oyvind Tafjord, Nathan Lambert, Yuling Gu, Shane Arora, Akshita Bhagia, Dustin Schwenk, David Wadden, Alexander Wettig, Binyuan Hui, Tim Dettmers, Douwe Kiela, Ali Farhadi, Noah A. Smith, Pang Wei Koh, Amanpreet Singh, and Hannaneh Hajishirzi. Olmoe: Open mixture-of-experts language models, 2024.

[59] 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.

[60] Arkil Patel, Satwik Bhattamishra, and Navin Goyal. Are nlp models really able to solve simple math word problems? In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2080–2094, 2021.

[61] Bo Peng, Eric Alcaide, Quentin Anthony, Alon Albalak, Samuel Arcadinho, Stella Biderman, Huanqi Cao, Xin Cheng, Michael Chung, Matteo Grella, et al. Rwkv: Reinventing rnns for the transformer era. *arXiv preprint arXiv:2305.13048*, 2023.

[62] Siva Reddy, Danqi Chen, and Christopher D. Manning. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266, 2019.

[63] Subhro Roy and Dan Roth. Solving general arithmetic word problems. *arXiv preprint arXiv:1608.01413*, 2016.

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

[65] Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. Social iqa: Commonsense reasoning about social interactions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4463–4473, 2019.

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

[67] Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158. Association for Computational Linguistics, 2019.

[68] Omkar Thawakar, Ashmal Vayani, Salman Khan, Hisham Cholakkal, Rao Muhammad Anwer, Michael Felsberg, Timothy Baldwin, Eric P. Xing, and Fahad Shahbaz Khan. Mobillama: Towards accurate and lightweight fully transparent gpt, 2024.

[69] Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *arXiv preprint arXiv:2305.14975*, 2023.

[70] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[71] Artem Vazhentsev, Akim Tsvigun, Roman Vashurin, Sergey Petrakov, Daniil Vasilev, Maxim Panov, Alexander Panchenko, and Artem Shelmanov. Efficient out-of-domain detection for sequence to sequence models. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1430–1454, 2023.

[72] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in neural information processing systems*, pages 24824–24837, 2022.

[73] Gwenyth Portillo Wightman, Alexandra Delucia, and Mark Dredze. Strength in numbers: Estimating confidence of large language models by prompt agreement. In *Proceedings of the 3rd Workshop on Trustworthy Natural Language Processing (TrustNLP 2023)*, pages 326–362, 2023.

[74] Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International Conference on Machine Learning*, pages 38087–38099. PMLR, 2023.

[75] Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*, 2023.

[76] Miao Xiong, Zhiyuan Hu, Xinyang Lu, YIFEI LI, Jie Fu, Junxian He, and Bryan Hooi. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. In *The Twelfth International Conference on Learning Representations*, 2024.

[77] Haoran Xu, Baolin Peng, Hany Awadalla, Dongdong Chen, Yen-Chun Chen, Mei Gao, Young Jin Kim, Yunsheng Li, Liliang Ren, Yelong Shen, et al. Phi-4-mini-reasoning: Exploring the limits of small reasoning language models in math. *arXiv preprint arXiv:2504.21233*, 2025.

[78] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie

Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zhihao Fan. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

[79] Gal Yona, Roee Aharoni, and Mor Geva. Can large language models faithfully express their intrinsic uncertainty in words? *arXiv preprint arXiv:2405.16908*, 2024.

[80] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2019.

[81] Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. Tinyllama: An open-source small language model. *arXiv preprint arXiv:2401.02385*, 2024.

[82] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

[83] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 2023.

[84] Zesen Zhao, Shuowei Jin, and Z Morley Mao. Eagle: Efficient training-free router for multi-llm inference. *arXiv preprint arXiv:2409.15518*, 2024.

[85] 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. In *Advances in Neural Information Processing Systems*, pages 46595–46623, 2023.

[86] Zhengping Zhou, Lezhi Li, Xinxi Chen, and Andy Li. Mini-giants:" small" language models and open source win-win. *arXiv preprint arXiv:2307.08189*, 2023.

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Last paragraph of Section 1

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: Appendix E

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Section 3.1

Guidelines:

- The answer NA means that the paper does not include experiments.

- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Abstract (`https://anonymous.4open.science/r/quodlibeta`)

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Section 3.1

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA]

   Justification: [NA]

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix D

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.

- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: All dataset used in this work are public available. This research has no violation of ML safety or human rights.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Appendix F

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [NA]

    Justification: All the code, data, and models in this work is opensourced with Apache-2.0 licence.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: [NA]

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: [NA]

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: Section 3.1 describes the usage of LLMs.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (https://neurips.cc/Conferences/2025/LLM) for what should or should not be described.