BUDGET-EFFICIENT LLM SELECTION VIA AUTOMATED SKILL PROFILING

Anonymous Authors[1]

ABSTRACT

Introduction. The surge in large language models (LLMs) has introduced a wide range of deployment options—from lightweight open-source models to high-performing proprietary APIs. While larger models typically yield better accuracy and reasoning, they come with significantly higher monetary and energy costs. This raises a core systems challenge: how can we select the most suitable LLM for a given task, under real-world constraints such as latency, energy, or budget?

Benchmark metrics alone are insufficient to guide this decision. Choosing the most accurate model often ignores which capabilities are required, why failures occur, or whether a cheaper model might suffice. We introduce **BELLA** (Budget-Efficient LLM Selection via Automated Skill Profiling), a system that decomposes LLM outputs into interpretable skills and weaknesses, builds structured model-task-skill matrices, and recommends models that maximize utility within resource limits. While we demonstrate BELLA on financial reasoning tasks, its architecture generalizes to any domain exhibiting cost-performance tradeoffs.

Background and Related Work. Conventional LLM evaluation pipelines emphasize aggregate task metrics (e.g., accuracy, F1) or compute costs (e.g., FLOPs, latency), but provide limited insight for deployment decisions. Recent profiling tools offer finer granularity: EvalTree (Zeng et al., 2025) builds hierarchical skill trees but assumes one skill per instance; Skill-Slices (Moayeri et al., 2024) cluster rationales to derive skill-level performance but require detailed reasoning traces. QualEval (Murahari et al., 2024) and CheckList (Ribeiro et al., 2020) provide test-based diagnostics but lack per-instance attribution and do not incorporate deployment-time constraints like cost or latency.

Separately, model routing frameworks like GraphRouter (Feng et al., 2025), AS-LLM (Wu et al., 2024), and ModelSwitch (Chen et al., 2025) improve efficiency via learned policies or model agreement, and in some cases explicitly factor in cost. However, they do not reason over skill-level capabilities or offer interpretable insight into why a model is selected for a given task. BELLA complements these systems by aligning required and demonstrated skills while optimizing for real-world resource limits—enabling cost-aware selection grounded in interpretable, per-instance capability profiling.

Methodology. BELLA is a modular pipeline for skill-based model selection under deployment constraints. It consists of five stages:

1. **Benchmarking.** We evaluate a suite of LLMs on diverse multi-skill reasoning tasks, collecting standardized performance metrics (e.g., accuracy, F1) and operational costs (e.g., API usage, latency, energy). These serve as the empirical foundation for downstream profiling and selection.
2. **Skill Profiling via Critic LLM.** For each model-task-instance, a critic LLM is prompted with the input, reference output, and model output. It returns a structured list of demonstrated and missing skills, grounded in the model's reasoning trace. This yields a fine-grained, per-instance capability profile.
3. **Skill Clustering and Canonicalization.** We embed and cluster all annotated skill/weakness phrases to derive a compact, interpretable set of $\sim$15–20 canonical skills. Each instance is relabeled with binary indicators for skill presence and impact on outcome.
4. **Skill-Aware Model Selection.** For a new task, BELLA infers the required skills (via prompting or user input), filters models that cover those skills, and applies multi-objective optimization to select one that maximizes performance within budget constraints.
5. **Evaluation.** We evaluate BELLA using **leave-one-task-out cross-validation**. For each held-out task, we infer its required skills, apply BELLA's selection process, and compare performance and cost against oracle and naive baselines. We also test alternative skill profilers (e.g., EvalTree) to assess robustness to representation quality.

BELLA enables plug-and-play skill extraction, interpretable model routing, and cost-aware selection—bridging the gap between benchmark metrics and deployment-ready decisions. Its modular design supports transparent, efficient use of LLMs across diverse tasks, models, and constraints, making it a deployable, domain-agnostic solution for cost-efficient inference.

REFERENCES

Chen, A. et al. Do we truly need so many samples? multi-llm repeated sampling efficiently scales test-time compute. *arXiv preprint arXiv:2504.00762*, 2025.

Feng, T., Shen, Y., and You, J. Graphrouter: A graph-based router for llm selections. *OpenReview*, 2025. URL https://openreview.net/forum?id=eU39PDsZtT.

Moayeri, M., Gehrmann, S., and Beutel, A. Unearthing skill-level insights for understanding trade-offs of foundation models. *arXiv preprint arXiv:2410.13826*, 2024.

Murahari, V., Zhang, T., and Wang, Y. Qualeval: Automatic capability profiling and skill discovery for llms. In *NAACL*, 2024.

Ribeiro, M. T., Wu, T., Guestrin, C., and Singh, S. Beyond accuracy: Behavioral testing of nlp models with checklist. In *ACL*, 2020.

Wu, X., Zhong, Y., Wu, J., and Tan, K. C. As-llm: When algorithm selection meets large language model. *OpenReview*, 2024. URL https://openreview.net/forum?id=l7aD9VMQUq.

Zeng, Z., Wang, Y., Hajishirzi, H., and Koh, P. W. Evaltree: Profiling language model weaknesses via hierarchical capability trees. *arXiv preprint arXiv:2503.08893*, 2025.