Surpassing LLMs for tool calling with SLMs

Akshat Jain (B22CS096) Ishaan Pandey (B22CI017) Tejas Gupta (B22CS093)

Indian Institute of Technology - Jodhpur

September 21, 2025

Abstract

While Large Language Models (LLMs) exhibit powerful zero-shot capabilities, their significant computational overhead presents a barrier to widespread adoption. This research investigates the efficacy of Parameter-Efficient Fine-Tuning (PEFT) on Small Language Models (SLMs) for tool-calling tasks. We will conduct a comparative analysis of a base SLM, the same SLM after fine-tuning, and a general-purpose LLM. We hypothesize that the specialized SLM will demonstrate superior performance on in-domain tasks compared to its base model, while achieving a competitive performance-per-parameter ratio against the LLM, validating a resource-efficient approach to building capable AI agents.

Methodology

1. **Experimental Setup:** The project will utilize Python with PyTorch and the Hugging Face ecosystem for model handling and training. All experiments will be designed to be reproducible on publicly available platforms like Google Colab.
2. **Model Selection:** Our study involves three subject models:
   - **Control Group (Base SLM):** An unmodified, pre-trained SLM such as 'Phi-3-medium' to establish baseline zero-shot performance.
   - **Experimental Group (Fine-tuned SLM):** The same 'Phi-3-medium' model, fine-tuned using PEFT.
   - **High-Performance Baseline (LLM):** A larger model, 'Llama-3-8B-Instruct', for a comparative benchmark against a capable, generalist model.
3. **Fine-tuning Protocol:** We will employ the LoRA technique for PEFT on the API-Bank dataset. The primary objective is to instill the model with the ability to map natural language intents to structured, executable API calls in JSON format.
4. **Quantitative Evaluation:** All three models will be systematically benchmarked on a held-out test set from the BFCL. This ablation study will allow us to precisely measure the impact of fine-tuning.
5. **Analysis of Results:** We will analyze the statistical significance of the performance delta between the control and experimental groups and characterize the trade-offs between the specialization of our fine-tuned SLM and the generalization of the baseline LLM.

Expected Outcomes

- **Empirical Validation of Specialization:** To provide quantitative evidence that fine-tuning significantly enhances an SLM's tool-calling capabilities over its foundational zero-shot abilities.
- **Efficiency Benchmarking:** To demonstrate that a specialized SLM can offer a superior performance-per-parameter trade-off, achieving results competitive with a larger LLM on in-domain tasks.
- **Contribution to Reproducible Research:** To produce a well-documented framework, including open-source scripts, enabling other researchers to replicate and build upon our findings.
- **Dissemination of an Open-Source Artifact:** To potentially release our final fine-tuned model weights to the Hugging Face Hub, providing a tangible contribution to the community.

Databases and Evaluation Metrics

Databases:

- **API-Bank:** Utilized as the primary training corpus due to its diverse API set and its emphasis on multi-turn dialogues, which is critical for developing contextual understanding.
- **Berkeley Function Calling Leaderboard (BFCL):** Employed as the primary evaluation suite to test for functional correctness and precision in real-world scenarios.

Evaluation Metrics:

- **Call Accuracy:** A strict metric for functional correctness, calculated as the percentage of perfectly formulated API calls.
- **Decision Accuracy:** Evaluates the model's reasoning by measuring its correctness in deciding whether tool invocation is necessary.
- **Hallucination Rate:** The frequency of generating non-existent tools or parameters, a critical metric for assessing model reliability and trustworthiness.
- **Computational Cost:** The resources required for the fine-tuning process, measured in GPU hours, to quantify the model's training efficiency.

Conceptual Diagram

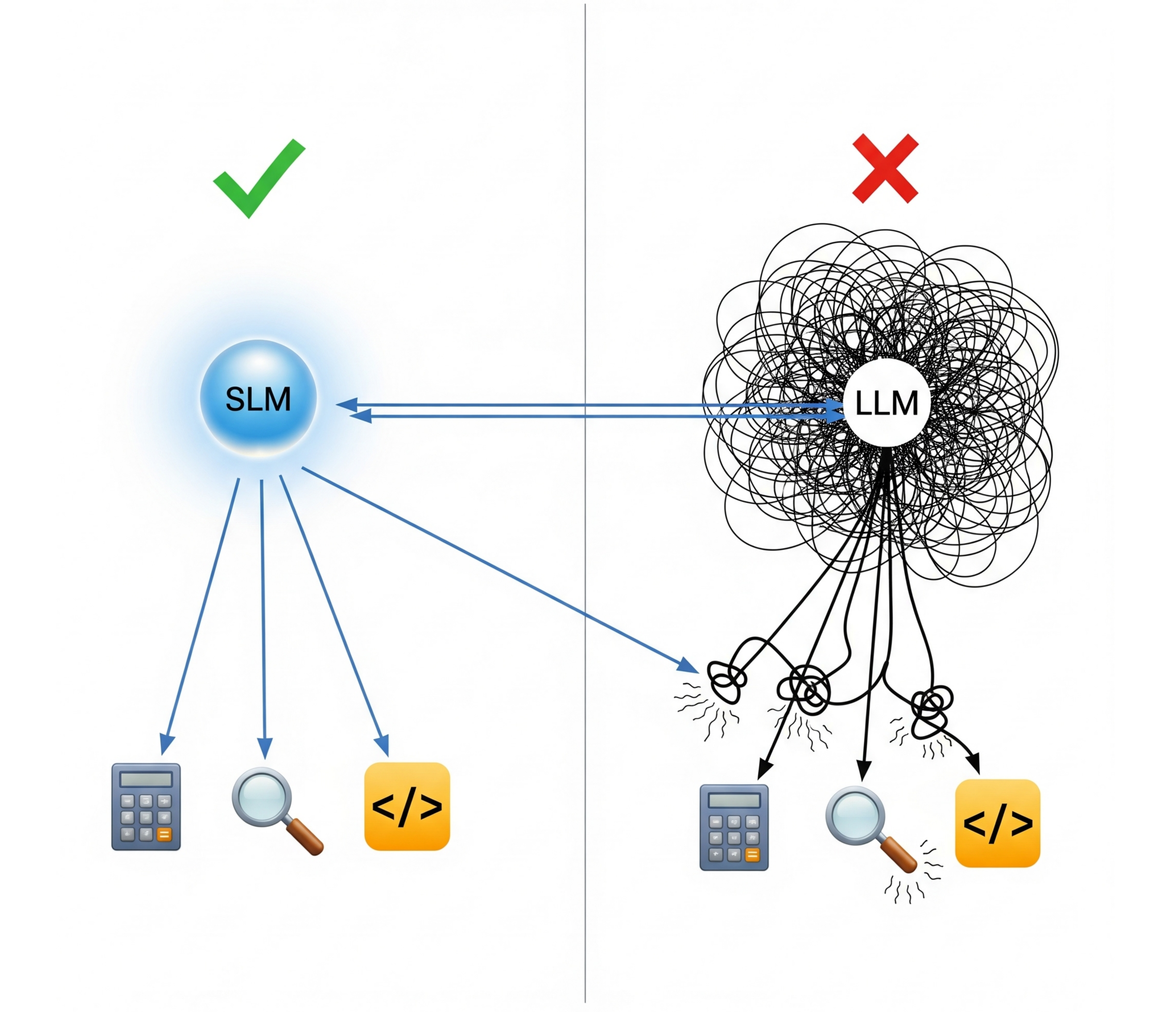

Figure: A diagram illustrating an efficient SLM successfully using a variety of tools, while a larger, more complex LLM struggles with the same tasks, highlighting the SLM's superior performance and efficiency.

References

1. Schick, T., et al. (2023). *Toolformer: Language Models Can Teach Themselves to Use Tools.* arXiv:2302.04761.
2. Patil, S., et al. (2023). *Gorilla: Large Language Model Connected with Massive APIs.* arXiv:2305.15334.
3. Li, M., et al. (2023). *API-Bank: A Comprehensive Benchmark for Tool-Augmented LLMs.* EMNLP.