Device-Aware Distillation of Small Language Models for Efficient Edge Intelligence

Vinamra Sharma*, Danilo Pau†, José Cano*
*University of Glasgow, Scotland, UK †STMicroelectronics, Italy

The rapid development of large language models (LLMs) has enabled strong reasoning and decision-making performance across a wide range of domains [1] and complex system tasks such as Design Space Exploration [2]. However, their direct deployment on mobile and edge hardware remains constrained due to substantial memory demand, computational cost, inference latency, and continuous dependence on cloud infrastructure. These limitations are vital for low-latency and privacy-sensitive applications where local intelligence is required. Although small language models (SLMs) provide a practical deployment path, their effectiveness depends on task adaptation and the alignment with the target device capabilities.

Recent studies [3], [4] suggest that SLMs can provide an effective alternative to LLMs for specialized applications. In [5], authors introduced SLaM, an automated framework for evaluating open-source SLMs against LLM services using response quality, latency, reliability, and cost; where several SLMs achieved response quality close to GPT-4 while delivering more consistent latency and reducing operational cost by 5× to 29×. Also, post-training and knowledge distillation has shown that SLMs can approach or exceed the performance of untuned LLMs [6], [7].

Despite these advances, a major gap remains between the demonstrated capability of SLMs and their practical deployment on heterogeneous edge hardware. Existing studies [5], [8] typically evaluate pre-trained SLMs as a replacements for LLMs, or focus independently on model compression and post-training quality improvements. However, effective edge deployment requires considering multiple factors, including task-specific supervision, data availability, base model suitability, resource constraints of the target device, quantization strategy, and the trade-offs between accuracy and performance. Currently, there are limited works [9] that unify these stages into a reproducible workflow that can transform a general-purpose SLM into a domain-specialized hardware-aware model; which becomes our initial motivation.

This paper presents an end-to-end framework that enables compact open-source language models(<=7B size) to be specialized for edge deployment through a knowledge distillation driven workflow. The framework automatically generates task specific supervised fine-tuning data using a stronger cloud-hosted teacher model, recommend the user a suitable adaptation strategy based on target hardware. Users specify the application in natural language and selects hardware target, after which the framework recommends a suited fine-tuning methodology, and quantization. Once validated, the framework then starts generating a balanced fine-tuning data via the teacher LLM and then starts with fine-tuning and quantization to finally provide a deployment ready version.

Validation was conducted using the public Healthcare Fraud Detection dataset [10]. The distilled and quantized TinyLlama-1.1B model generated by the framework achieved 95% task accuracy on the Raspberry Pi 5, compared to it's 62% accuracy pre fine-tuned, while the cloud-hosted Kimi K2.6 teacher model achieved 97% accuracy under the same evaluation settings. To evaluate deployment scalability and hardware-aware edge execution, we deployed a quantized Qwen3 model Q3_K_M on the STM32MP2 embedded MPU using llama.cpp. The evaluation demonstrated successful fully local CPU-only inference with 3.2 tokens/s prompt throughput and 2.1 tokens/s generation throughput, while maintaining runtime memory usage below 700 MB without swap utilization. These early results show the potential of SLMs to attain application-specific task accuracy close to LLMs when distilled and fine-tuned with hardware awareness.

Acknowledgment

This work was partially supported by the EU Project dAIEDGE (GA Nr 101120726) and the Innovate UK Horizon Europe Guarantee (GA Nr 10090788).

References

[1] J. Cheng *et al.*, "Realm: A dataset of real-world llm use cases," in *ACL*, 2025.
[2] V. Sharma *et al.*, "LLM-driven design space exploration of FPGA-based accelerators," 2026.
[3] M. S. Y. Alassan *et al.*, "Comparison of open-source and proprietary llms for machine reading comprehension: A practical analysis for industrial applications," *arXiv*, 2024.
[4] G. Bai *et al.*, "Beyond efficiency: A systematic survey of resource-efficient large language models," *arXiv*, 2024.
[5] C. Irugalbandara *et al.*, "Scaling down to scale up: A cost-benefit analysis of replacing openai's llm with open source slms in production," in *ISPASS*, 2024.
[6] M. Rang *et al.*, "Revealing the power of post-training for small language models via knowledge distillation," *arXiv*, 2025.
[7] M. Ballout *et al.*, "Efficient knowledge distillation: Empowering small language models with teacher model insights," in *ICANLIS*, 2024.
[8] W. Ye *et al.*, "Select to think: Unlocking slm potential with local sufficiency," *arXiv*, 2026.
[9] N. Dhar *et al.*, "An empirical analysis and resource footprint study of deploying large language models on edge devices," in *ACMSE*, 2024.
[10] N. Abbas, "Healthcare fraud detection dataset," 2026. [Online]. Available: https://www.kaggle.com/datasets/nudratabbas/healthcare-fraud-detection-dataset