Project 1: Profiling and OS/Runtime Co-Design for Processing-in-Memory (PIM) in Mobile Generative AI Workloads

Objective:

Investigate how memory access patterns impact the performance of Generative AI (GenAI) and Large Language Models (LLMs) on mobile devices, and explore how the OSor runtime can be redesigned to better utilize PIM accelerators.

Description:

This project will profile GenAI/LLM workloads running on mobile platforms to identify memory bottlenecks (e.g., frequent page faults, irregular access patterns). Using tools such as our QEMU-based tool, students will measure and analyze memory access behavior. Students will then write a report summarizing their insights. Based on insights, the project will propose modifications to the Linux kernel (e.g., new memory management schemes) or runtime system (e.g., scheduling policies) to better exploit PIM capabilities as future work.

Expected Outcome:

- A detailed characterization of memory access patterns for GenAI workloads.
- Identification of bottlenecks that prevent effective PIM utilization.
- Prototype OS/runtime extensions that reduce overhead and accelerate GenAI execution by better exploiting PIM capabilities (optional or for future work).

Project 2: Revisiting OS to Support Emerging Near-Data Processing Accelerators

Objective:

Analyze limitations in today's OS design that hinder adoption of near-data computing (e.g., PIM) for mobile GenAI workloads, and extend the OS to support these new hardware capabilities.

Description:

This project will survey recent papers for PIM, particularly processing-using-memory, with a focus on their requirements (e.g., data layout, instruction support, coherence). Students will identify current Linux kernel designs that conflict with these requirements or assumptions (e.g., page granularity, uniform caching policies). They will write a report summarizing their observations and proposing some preliminary solutions that relax these assumptions and

align the OS with accelerator needs. This project will implement kernel-level changes—such as alternative page table structures, new system calls, or modified memory allocators—based on the students' reports as future work.

Expected Outcome:

- A systematic study of mismatches between OS design and requirements/assumptions of near-data computing hardware.
- Generate some insights with concrete data support to show the mismatches.
- Concrete kernel modifications that remove limitations (e.g., enabling special instructions, supporting accelerator-friendly data layouts) (optional or for future work).
- Demonstration of improved performance or efficiency on representative GenAI workloads (optional or for future work) .