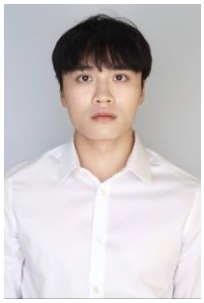
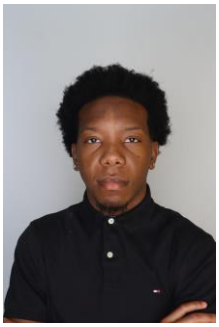
IP-10 AFB LLM Compression Trade-off Evaluator

CS 4850 - Section 03 – Fall 2025

Aug 26, 2025

 Ryan Tran Developer/Documentation	 Aldi Susanto Developer/Documentation/Team Lead	 Thomas Graddy Developer/Documentation	 Pranav Kartha Developer/Documentation

Team Members:

Name	Role	Cell Phone / Alt Email
Ryan	Documentation/Development	678-670-9868 Concepting@protonmail.com
Aldi	Team Lead/Documentation/Development	404-834-9304 aldisusanto01@gmail.com
Thomas	Documentation/Developmen	229-661-5041 Thomas.graddy3@gmail.com
Pranav	Documentation/Developmen	943-268-4909 pranav.kartha1950e@gmail.com
Sharon Perry	Project Owner or Advisor	770-329-3895 Sperry46@kennesaw.edu
Taylor Cuffie	Academic Program Support Specialist	470-578-5760 tcuffie1@kennesaw.edu

Collaboration Tools:

Communication — Discord (Primary) Teams, Cellphones (Call/Text), KSU Email/Alt Email
Collaboration — Discord (in between weekly status meetings on Teams !)

Abstract:

Large Language Models are increasingly applied to code-generation tasks. We have tools like Claude Code, Cursor, Lovable, etc. But their substantial size and computational requirements limit efficiency in real-world deployment. In this industry project, sponsored by Robins AFB – AFSC/EN, our team will be investigating the trade-offs between LLM compression and inference quality, with a focus on code-related queries. We will evaluate compression methods such as pruning, quantization, and knowledge distillation, measuring their impact on accuracy and performance through metrics like BLEU and Code-BLEU.

In addition to benchmarking compressed models, in this project, we aim to develop a dynamic query routing prototype that intelligently allocates tasks between full-scale and compressed LLMs based on query complexity. Our results will be visualized through an interactive dashboard, highlighting performance trends, resource savings, and operational feasibility.

The outcomes of this work will provide practical insights into optimizing LLM deployment for code-generation tasks, offering pathways to reduce computational costs while maintaining inference quality. By delivering benchmarking scripts, evaluation reports, and deployment recommendations, this project contributes to advancing efficient and scalable AI solutions for mission-critical environments.

Platform:

The LLM Compression Trade-off Evaluator will be developed using a modern AI/ML and full-stack technology stack. The primary experimentation and model compression work will be carried out in Python with frameworks such as PyTorch and the Hugging Face Transformers library. These provide robust tools for implementing and fine-tuning compression techniques including pruning, quantization, and knowledge distillation.

For deployment and testing, the system will utilize Docker containers to ensure reproducibility and portability across environments. Experiments will be run both locally and, where applicable, in cloud environments such as AWS, GCP, or Azure, allowing scalable benchmarking of compressed and full-size LLMs.

Visualization and reporting will be handled through an interactive web-based dashboard built with Streamlit or Dash, enabling clear presentation of evaluation metrics such as Code-BLEU scores, accuracy trade-offs, and resource savings. Version control and collaboration will be managed using GitHub/GitLab, ensuring structured development and team coordination.

This platform stack provides the flexibility to conduct rigorous experimentation, build a functioning proof-of-concept for dynamic query routing, and deliver interactive, data-driven insights into the efficiency of LLM deployment.

STATEMENT OF PARTICIPATION

By submitting this assignment, I acknowledge that I will participate in all meetings, communications, deliverables and other tasks necessary to complete the project. If I do not participate, I understand that Professor Perry will meet with me to remedy the situation.