Final: Build, Train, and Imagine with LLMs

Due Date: Friday, December 19, 2025, at 12:00 AM ET (midnight)

- No extensions due to the university's grading freeze period.

🔑 Key Info

- Up to 4 people per team (*individual submissions also allowed*).
- Projects may focus on **applications**, **model training/adaptation**, or any topic that meaningfully involves Large Language Models (LLMs) - essentially, anything you would be proud to showcase on your resume.
- Your goal is to demonstrate **creativity**, **insight**, and **technical depth** - anything that will create the "WOW" moment.

📃 Submission

You will submit the following 3 items:

- 10-minute Video Presentation
  - Introduce the motivation, goal, and novelty of your project.
  - Show a demo or prototype in action (recorded or live demo).
  - Explain your technical approach and stack (model choice, data, tools, workflow).
  - Briefly describe team roles - who did what.
  - Note: Focus on **clarity**, **storytelling**, and **takeaways**.
- Project Report (up to 8 pages)
  - Format: PDF (no code, only technical report text and diagrams).
  - Include, but are not limited to, the following:
    - Project name, and team members
    - Abstract
    - Introduction & Motivation
    - Related Work (what inspired your approach)
    - Methodology / System Design (describe pipeline, architecture, or experiment)
    - Evaluation / Results (quantitative or qualitative)
    - Discussion & Limitations
    - Conclusion & Future Work
  - Note: Think from an **elevator-pitch** perspective - how would you convince someone of your idea in a few minutes? Use diagrams, flowcharts, and tables to **tell your story visually** and make your report clear, structured, and compelling.
  - Note: Name your files as: "*TeamName_Final_Project.pdf*" and include your team members' names in the report.
- Code Repository Link
  - Provide a working GitHub, or Hugging Face repo with clear setup instructions.

💡 Project Reference

You can explore any of the following - or better yet, invent your own!

These references are only meant to **spark ideas, not limit your creativity**.

Think big, think bold - surprise your audience with something useful, artistic, or simply mind-blowing.

Application-focused

Design innovative end-user tools that make LLMs practical or delightful:

- Personalized AI Agent - context-aware agent that remembers, reasons, and plans.
- AI Tutor / Coach - interactive mentor that explains, quizzes, or adapts to learning style.
- Creative Generator - storytelling, script-writing, or meme-composer with multimodal flair.
- Collaborative Writing Partner - co-author that edits and critiques in real time.
- Knowledge Companion - answers questions using domain-specific corpora or APIs (law, medicine, finance, travel).
- Productivity Tools - summarizer, task planner, code explainer, or auto-email writer.
- AI for Good - accessibility aid, language learning assistant, or environmental chatbot.

Model-focused

Work with open-source models to enhance efficiency, specialization, or robustness:

- LoRA / Adapter / PEFT exploration - efficient training on domain-specific data.
- Quantization and Distillation - make large models lighter and faster.
- Instruction or dialogue tuning - align behavior to human preferences.
- Compare scaling laws - small vs large model performance under same tasks.

Research-focused

Push boundaries of what LLMs can reason, remember, or imagine:

- Chain-of-Thought refinement - teach models to verify or debate their reasoning.
- Long-context modeling - summarize or answer across multi-document input.
- Mixture of Experts (MoE) - modular reasoning system that routes tasks to specialists.
- Multi-agent collaboration - several LLMs coordinating, arguing, or teaching each other.
- Cross-lingual or cross-domain adaptation - study transfer learning between languages, modalities, or specialized domains.
- Safety and alignment experiments - explore prompt engineering, red-teaming, or refusal-behavior tuning to reduce harmful outputs.
- Memory-augmented architectures – design external memory or vector-database modules for persistent recall and reasoning continuity.

📊 Grading Rubric

Category	Percentage	Description
Creativity & Novelty	25	Original idea or creative use of LLMs
Technical Depth	25	Quality of design, model, or analysis
Clarity & Communication	20	Clear presentation and storytelling
Demo Quality	20	Functionality, polish, and impact
Team Collaboration	10	Balanced effort and contribution clarity

🧩 Tips for Success

- Start early - explore and iterate!
- Think storytelling: make your video and demo engaging.
- Do not chase massive scale - a small but well-executed demo is better than an unfinished grand idea.
- Creativity > compute budget - Please check the Ed threads for updates, as we are actively applying for additional GPU/TPU resources for the teams.