CSI-4130/5130: Artificial Intelligence - Course Project

Overview: AI in the Era of Generative Models

The field of Artificial Intelligence is undergoing a transformation, driven by the remarkable capabilities of Generative AI and Large Language Models (LLMs). This project is your opportunity to apply cutting-edge AI to solve a real-world problem that you are passionate about.

This project is completely open. You are encouraged to be creative and ambitious. You can reproduce a research paper, participate in a Kaggle challenge, build upon open-source libraries, or develop a completely new idea. The core requirement is to apply AI skills to solve an interesting problem. Your grade will be based on the novelty and quality of your solution, the technical depth of your work, and the potential impact of your project.

This year, we are especially encouraging projects that leverage modern AI paradigms. You might consider one of the following paths:

1. **Application Development with LLM APIs:** Design and build a novel application by integrating with existing state-of-the-art models from providers like **OpenRouter**, **OpenAI**, **Google AI**, **Anthropic**, **xAI**, or other platforms.
2. **Model Fine-Tuning:** Take a powerful open-source pre-trained model (e.g., Gemma, Mistral) and fine-tune it on a specialized dataset to excel at a specific task.
3. **Novel Model or Algorithm Development:** For advanced students, propose and implement a new model architecture, a novel training technique, or an innovative algorithm.
4. **Comprehensive Model Evaluation:** Conduct a rigorous analysis and comparison of different generative models or prompting techniques on a specific real-world problem.

To succeed, you will need to define a clear problem, collect or curate a relevant dataset, design your approach, implement your system, and rigorously evaluate the results. Be mindful that training or extensive API usage can be resource-intensive. Plan accordingly and leverage resources like Google Colab, Kaggle, and available university computing clusters.

Collaboration Policy

You may work individually or in teams of up to **three** people. Team-based projects are expected to have a proportionally larger scope. In a team, you must ensure every member makes a significant and distinct contribution. At the end of the project, each team member will confidentially report their own contributions and those of their teammates.

Honor Code & Attribution

You may consult any papers, books, online references, or publicly available code for ideas. You are free to use open-source libraries and models, but you **must** attribute them properly.

Directly copying code, results, or text without proper citation is a serious violation of academic integrity. Your GitHub repository's README.md and final report must have a dedicated section for citations and acknowledgements.

If you are combining this with a project from another course, you must get prior permission from all instructors involved and clearly delineate the work done for this course.

Submissions, Grading, and Important Dates

Your final grade for the project will be based on the following components:

- **GitHub Progress (10%):** Assessed via your repository's commit history.
- **Final Project Report (30%):** The written summary of your work.
- **In-Class Presentation (60%):** Your presentation and demo to the class, which will include peer grading.

Important Dates:

- **GitHub Repository Link Submission:** Due 11:00 PM, Sunday, Oct 26, 2025
- **Final Project Submission (Report & Final Repo State):** Due 11:00 PM, Monday, Dec 1, 2025
- **In-Class Presentations:** Tuesday, Dec 2 & Thursday, Dec 4, 2025

Project Development & GitHub Repository (Due Oct 26)

Instead of a formal proposal, you will create a public GitHub repository at the start of your project. This is an essential skill for any developer or researcher.

1. **Create a Repository:** This will be your project's home for all code, documentation, and progress.
2. **Initial README.md:** Your initial commit should include a README.md file that briefly outlines your problem statement, proposed method, and data sources (similar to a mini-proposal).
3. **Submit Your Link:** Submit the link to your public GitHub repository on Moodle by the deadline.
4. **Show Consistent Progress:** Meaningful, regular commits are expected. Your commit history should tell the story of your project's development and help clarify each team member's contribution.

Final Project Report (Due Dec 1)

While the presentation and demo are the main focus, a written report is still required to document your work. It should be structured like a short academic paper.

- **Title and Author(s)**
- **Abstract:** A brief summary of the project's problem, approach, and key results.

- **Introduction:** Motivate the problem, explain its significance, and provide a high-level overview of your approach.
- **Related Work:** Discuss relevant work and how your project differs.
- **Data:** Describe your dataset, its source, and any preprocessing steps.
- **Methods:** Detail your technical approach, including models, frameworks, and algorithms used.
- **Experiments & Results:** Present your evaluation. For generative tasks, include both quantitative metrics and a strong qualitative analysis with examples.
- **Conclusion:** Summarize your accomplishments, limitations, and potential future work.
- **Citations & Acknowledgements:** Properly credit all sources.

Submission: Update your GitHub repository with the final report (PDF) and all source code. Ensure the README.md is updated with clear instructions on how to run your project.

Project Presentation (Dec 2 & 4)

You will present your project to the class. A live demo is highly encouraged! Each team member must speak.

Your presentation grade will be a weighted average of peer feedback and instructor grading. You are expected to provide thoughtful and constructive feedback on your classmates' projects.

A suggested presentation structure:

1. **Problem Statement:** What problem did you tackle and why is it important?
2. **Your Solution:** Briefly describe your technical approach and key innovations.
3. **Results & Demo:** Show us what you built! Present key findings and provide a live demo.
4. **Broader Impact & Future Work:** What did you learn, what are the limitations, and how could your work be extended?