

CS5302/EE519 - Language and Speech Processing

Project Manual

Overview

The overarching aim of this project is to harness the potential of Generative Artificial Intelligence in crafting a tool or application that delivers tangible benefits to the average individual. Your task includes deploying the developed system, whether it takes the form of a website, mobile application, or even a Google Extension. Operating within groups of 5-6 members, each team will receive guidance from two TAs - one serving as the primary point of contact, and the other offering secondary support.

Within the realm of Generative AI, the spectrum of possibilities is vast and encompasses technologies beyond those covered in lectures. This grants you the freedom to explore Large Language Models such as GPT-4 and LLaMA, Text-to-Image models like Stable Diffusion, Text-to-Speech, and Speech-to-Text models, as well as interactive Agent.

Deliverables & Grading

- Project proposal report [15%]
- Completeness of the project [40%]
- Project final report write-up [25%]
- GitHub Repository [5%]
- Posters [5%]
- Final Evaluation [10%]
- Bonus: doing something special and unique [extra +10% within the project]

Project Proposal (Tentative Deadline: 26th February)

The proposal document will serve as a breakdown of the project, based on which we will make our expectations and base your grading off.

You should prepare a simple document, no more than 4 pages, covering the following points:

- **Problem Motivation:** What is the problem you've identified? Why is it relevant?¹ What is the potential impact of a solution to this problem?
- **Goal of the Project:** What is the objective of your project? What angle(s) of the identified problem does it seek to tackle? For what audiences is your project aimed at? If there are similar works to what you aim to do, how are you making your work distinct?
- **Solution Approaches:** How do you aim to tackle the problem? What types of models are you using? Do you aim to focus more on proprietary models or open-source models? What techniques do you aim to use (e.g. RAG, fine-tuning, training from scratch), and how do you justify them? How large are the models you aim to use, and how do you justify this in terms of the problem scope? How do you aim to make your work practical in terms of usability for the end-user?

¹ It is not necessary to include references, though it would make your case stand out here

- **Rough Timeline:** What are the milestones you expect to achieve on a weekly or biweekly basis?

You should prepare this document and get it reviewed by your assigned TAs **before your submission**. It is very likely the project scope will be modified to account for fairness across all groups: this means possibly adding on more tasks to flesh out your work.

In the event you wish to revise aspects of your project, whether that be pruning some features, changing up the design, or altering the timeline, you are allowed to make these revisions by **March 18th**, only under the supervision of your assigned TAs.

Checkpoints

To ensure work does not accumulate towards the end, we will expect all groups to report their progress on a **biweekly basis**² to their assigned TAs. You will be expected to roughly stick to the timeline provided in your proposal.

Please create a private Slack group with **all** TAs - you need only contact the two assigned to your group.

Consistently failing to report on time can incur a penalty.

Final Report (Tentative Deadline: April 26th)

Your final report should be a comprehensive writeup, no more than 8 pages, covering the following points:

- **Implementation details:** What tools or systems did you use? How does it all work under the hood? How did you fine-tune your model (if applicable)? What datasets did you use (if applicable) - synthetic or bonafide, existing or collected? *Include a diagram* on how all the components fit together.
- **Deployment:** How did you deploy your project? What aspects were there to consider here?
- **Experiments and Iterations:** How did you iterate and make improvements through your project?
- **Reflections:** What worked well? What didn't? What would be the next step to improving your system? How could you have remedied what went wrong?
- **Demonstration of the project:** Include some diagrams covering the main usage of your application.
- **Individual Contributions:** Include one page (outside of the 8 required) stating the specific contribution(s) of each group member, with the acknowledgement of all team members.

You are required to use [this template](#) for your report.

You can find some good examples [here](#) - note that the format is different but the content is ideal.

²Once every 2 weeks.

GitHub

You are required to use [GitHub](#) for managing your code. It is very easy to use, and would allow you to collaborate much easier, for TAs to monitor each member's involvement in the project through their commit history, and for you to share your project with the rest of the world (if you wish)³.

5% of your grade in this project is based on the quality of your code, how nicely your codebase has been structured, and (mainly) the consistency with milestones being achieved through the development phase - this will be gauged mostly through the commit history - how consistently groups were working on the project throughout the semester and not leaving everything for the last moment.

Please create a repository and add both your TAs as collaborators (ask us for our usernames).

Posters & Final Evaluation

One important component of this project comes towards the end of the semester when you will be asked to display and present your projects. This means having to create your own Poster that summarizes your project nicely. This will also be where your final evaluation will take place.

Please look into [this link](#) for what goes into a nice poster.

Your evaluation will be conducted by the course staff and (possibly) external judges.

Advice

- Don't spend too much or too little time planning.
- Use AI tools like [GitHub Copilot](#), [ChatGPT](#), [Perplexity.ai](#) etc. to make your life easier.
- Delegate responsibilities and work in parallel.
- There's tons of API providers that offer extremely cheap options (for the right model) so you don't have to worry about breaking the bank. Some options include: [Anyscale](#) (which also offers \$10 of free usage!), [fireworks.ai](#), and [OpenAI](#).
- Projects in this domain do not require you to be very proficient with the technicalities. Many tools available to you abstract away the details and focus on letting you implement your idea fast. Do not be intimidated about this project requiring "a lot of work".
- Look into [📄 Resources Sheet](#) before starting if you have no experience with this domain.

³ A potential advisor/employer can be very impressed with a deployed app, a solid GitHub repository, and a good README. This is the gold standard.