

# Course Project Proposal Instruction for CSE 156 Winter 2025

Instructor: Lianhui Qin

Due at 11:59pm PT, Feb 6, 2025 (no late days)

## 1 Options for The Course Project

There are two options for the course project. The first option is to **conduct an open-ended research project about NLP**. As discussed in the first lecture, you may choose a topic in 1) core NLP, 2) applied NLP or interdisciplinary studies, or 3) product or startup. Alternatively, you can **investigate a well-established NLP task** (the “default project”).

### 1.1 Open-Ended Project - (Core / Applied) NLP Research

Students can choose to conduct an open-ended research project about NLP. Note that the scope of the project should fit the timeline of the course. So please avoid defining too open-ended research questions that might not lead to concrete results by the end of the course.

You can use your current research project as your final project. However, the project has to be very relevant to the class and well-scoped so that you reach the end of the quarter with a clear output. You’ll want to avoid finishing the quarter with too many open-ended, unanswered questions. Please specify the state of the project before the class began, and clearly distinguish your contributions from those of outside collaborators and mentors, if there were any involved. We’ll want to know what you accomplished during this quarter specifically. Projects that were already completed before the quarter started or very close to completion are NOT allowed.

### 1.2 Open-Ended Project - Startup & Product

This track is an opportunity to bridge theory and practice, demonstrating both your understanding of NLP and your ability to apply it in real-world contexts. In this track, you will adopt NLP techniques introduced in the course to build a **realistic, user-centric product** that addresses a concrete problem. The goal is to go beyond **basic API calls and prompt engineering**, demonstrating a deeper understanding of how NLP can be integrated into a product with practical value.

#### Expectations

- Your project should involve **substantial NLP work**, such as fine-tuning models, developing custom pipelines, domain-specific data engineering, or innovating new approaches.
- Since this is **not a software engineering course**, your project will not be graded on **frontend design or backend performance**, unless the **core NLP contribution** directly enhances these aspects (e.g., NLP-powered UI accessibility or backend optimization).
- Similar to conducting **literature reviews in research**, you must perform **market and competitive product analysis** to:
  - Identify **existing solutions** and analyze their **limitations or pain points**.
  - Clearly articulate how your **product improves upon existing alternatives** or addresses an **unmet need**.
- We encourage you to **prioritize innovation and usability**—aim to build something **novel** (i.e., a product that has not been built before or significantly improves upon existing solutions).

## Deliverables

Your final submission should include two key components:

### 1. Product Demo Video (3-5 minutes, non-technical)

- Demonstrate a **realistic use case** of your product that can be understood by users **without** a CS background.
- Highlight the **problem** your product solves and how NLP plays a role.
- The focus should be on **how users interact with the product**, rather than technical details.

### 2. Technical Report

- **Software architecture:** Explain how the different components of your system interact.
- **Core NLP techniques:** Describe the key NLP methods used (e.g., fine-tuning, retrieval-based systems, knowledge graphs, transformer-based models, etc.).
- **Implementation details:** Discuss model selection, data processing, training (if applicable), and performance trade-offs.
- **Market & competitive analysis:** Summarize existing solutions, their limitations, and how your approach differs.
- **Challenges & limitations:** Reflect on difficulties faced and potential areas for improvement.

## 1.3 The Default Project

Your goal is to investigate a well-established NLP task. **We suggest a few tasks below, but you can also pick another task that has well-documented benchmark performance.**

1. **DROP** [[LINK to the Benchmark](#)] is a QA dataset which tests comprehensive understanding of paragraphs. In this crowdsourced, adversarially-created, 96k question-answering benchmark, a system must resolve multiple references in a question, map them onto a paragraph, and perform discrete operations over them (such as addition, counting, or sorting).
2. **StrategyQA** [[LINK to the Benchmark](#)] consists of creative and diverse yes/no questions that require implicit reasoning steps. To solve questions in StrategyQA, the reasoning steps should be inferred using a strategy. To guide and evaluate the question answering process, each example in StrategyQA was annotated with a decomposition into reasoning steps for answering it, and Wikipedia paragraphs that provide evidence for the answer to each step. StrategyQA includes 2,780 examples, each consisting of a strategy question, its decomposition, and evidence paragraphs. Analysis shows that questions in StrategyQA are short, topic-diverse, and cover a wide range of strategies.
3. **SciFact** [[LINK to the Benchmark](#)] is a dataset of 1.4K expert-written claims, paired with evidence-containing abstracts annotated with veracity labels and rationales. Due to the rapid growth in the scientific literature, there is a need for automated systems to assist researchers and the public in assessing the veracity of scientific claims.
4. **ProtoQA** [[LINK to the Benchmark](#)] is a question answering benchmark dataset for prototypical commonsense reasoning. Given a question regarding some prototypical situations, such as “Name something that people usually do before they leave the house for work,” the goal of this benchmark is to test the commonsense reasoning capabilities of artificial intelligence systems to provide possible valid answers to the question. The training set is gathered from an existing set of questions played in a long-running international game show FAMILY-FEUD, and the hidden evaluation set is created by gathering answers for each question from 100 crowd-workers.

5. **OpenBookQA** [\[LINK to the Benchmark\]](#) is a new kind of question-answering dataset modeled after open book exams for assessing human understanding of a subject. It consists of 5,957 multiple-choice elementary-level science questions (4,957 train, 500 dev, 500 test), which probe the understanding of a small “book” of 1,326 core science facts and the application of these facts to novel situations. For training, the dataset includes a mapping from each question to the core science fact it was designed to probe. Answering OpenBookQA questions requires additional broad common knowledge, not contained in the book. The questions, by design, are answered incorrectly by both a retrieval-based algorithm and a word co-occurrence algorithm. Strong neural baselines achieve around 50% on OpenBookQA, leaving a large gap to the 92% accuracy of crowd-workers.
6. **Abductive Natural Language Inference (aNLI)** [\[LINK to the Benchmark\]](#) is a new commonsense benchmark dataset designed to test an AI system’s capability to apply abductive reasoning and commonsense to form possible explanations for a given set of observations. Formulated as a binary-classification task, the goal is to pick the most plausible explanatory hypothesis given two observations from narrative contexts.

**Your goal of the project is three-fold:**

1. **Gain a deep understanding of the task.** Survey around the task to understand the motivation of the task, how it impacted the broader NLP field, popular methods developed to solve this task, etc.
2. **Identify your favorite method that solves the task, and summarize what they did.** Describe why you like the method, why you think it solves the task neatly, what its limitations are, and what you would do if you were to improve their method.
3. **Implement a new solution to the task (i.e., a new method to solve the task) bounded by the computational budget you can access through the class.** Your solution could be based on previous methods that tackled the task, but you need to innovate new components on top of them. Given the limited computational resources you have in hand, you can aim to approach (and potentially beat) the leaderboard performance of the given task. However, your final grade on the project would **NOT** be penalized if the performance of your method does not approach the state-of-the-art performance, as long as you analyze the potential reasons that bound the performance of your method thoroughly (e.g., lack of computation resource, incomparable learning paradigms like supervised vs. unsupervised learning).

## 2 Project Proposal

Please use the [NeurIPS style](#) for your project proposal. Your proposal should be **no more than two** pages excluding references and include:

- Your **team members** and their **UCSD emails**.
- The **project type** that your team chose: Open-ended vs. Default.
- Specific information to provide for different types of projects:
  - **Open-Ended Project (Core NLP & Applied NLP)**
    1. 1-2 paragraphs summarizing the open-ended research project you plan to conduct. Clearly state your research question, its motivations, and why it’s important. Discuss why you are confident that the project’s scope will fit into the class.
    2. 1-2 paragraphs describing your research plan, including concrete steps of how you will accomplish your final project, with an estimated timeline.
    3. If you decide to use your current research project for your final project, please specify the state of the project before the class began, and how you plan to extend the project during the time of the course. Identify any external collaborators or mentors who are involved in the project, and clearly distinguish your contributions from theirs.

– **Open-Ended Project (Product & Startup)**

1. 1-2 paragraphs summarizing your startup or product idea. Clearly define the **problem** you aim to solve, its **importance**, and how NLP plays a role in addressing it. Identify the **target users** and discuss why this solution is valuable.
2. Conduct **market and competitive analysis**:
  - \* Identify existing solutions and analyze their **limitations or pain points**.
  - \* Explain how your product offers a **novel solution** or improves upon current alternatives.
3. 1-2 paragraphs describing your **technical approach**, including:
  - \* The **NLP techniques** you plan to use (e.g., fine-tuning models, retrieval-based systems, knowledge graphs, data curation and engineering, etc.).
  - \* The **datasets** or resources you will use or collect.
  - \* Any potential **technical challenges** and how you plan to address them.
4. A detailed **implementation plan** with a timeline:
  - \* Define your **Minimum Viable Product (MVP)** for the course timeline.
  - \* Define the **essential feature and requirement list** of your product (please consult GPT/Gemini/Claude/etc. to have a better understanding if you don't have much software engineering background).
  - \* Specify any **external dependencies or resources** needed (e.g., APIs, cloud computing).
5. Describe how you will evaluate the **success of your project**, including:
  - \* **NLP model performance metrics** (if applicable).
  - \* **Business or user engagement metrics** that reflect the product's impact.

– **Default Project**

1. The name of the task that you choose to investigate, along with a citation of the original paper where the benchmark was proposed.
2. An URL to the official benchmark of the task of your choice.
3. A few sentences describing why you pick the task.
4. 1-2 paragraphs describing your research plan, including concrete steps of how you are going to accomplish your final project, with an estimated timeline. You should describe your steps very specifically so that the course staff can provide concrete, useful feedback. For example, if you are going to identify prominent methods of the task, how or where do you plan to find such information?

## Acknowledgement

The final project instruction is primarily designed by Yichi Yang and Frederick Pi, and is partially adapted from the course materials from Yejin Choi.