Homework 4: Jailbreaking Large Language Models

CS 490: Natural Language Processing · **Spring 2026**

Due on: 04/20/2026 @ 11:59 PM (AoE)

Overview

In this assignment, you will investigate the robustness and safety of large language models (LLMs) under adversarial prompting, commonly referred to as **jailbreaking**. You will interact with a sequence of models hosted on `genai.rcac.purdue.edu`, where each model corresponds to a different difficulty level or architecture. Each model is instructed to protect a hidden password and refuse to reveal it directly.

Your goal is to systematically try different jailbreak strategies, observe how the models behave, and document your findings. You must submit a 2–3 page report using the ACL template. All figures must be created **digitally**. You must also include all prompts and model responses in the appendix.

Part 1: Threat Model and Attack Taxonomy

Before attempting any jailbreaks, you must define a threat model and organize your attacks into categories. The threat model consists of the capabilities of the attacker (e.g., how can the attacker interact with the system? what information is available to the attacker?), the harm caused if the attack succeeds (e.g., disclosure of the hidden password), the target system being attacked, and the existing mitigation mechanisms that limit the attacker's capabilities (e.g., what are existing safeguards that aim to prevent or discourage attacks?).

Examples of Jailbreak Attack Strategies

To get you started, here are several categories of attack strategies you may explore:

- **Direct Prompt Injection:** Ask the model to ignore previous instructions, override its safety constraints, or reveal hidden information.
- **Role-Playing / Persona Attacks:** Instruct the model to act as a different agent, system debugger, evaluator, or unrestricted assistant.
- **Multi-Turn Escalation:** Use a sequence of prompts to gradually steer the model into an unsafe or policy-violating state.
- **Encoding / Transformation Attacks:** Ask the model to output information in a transformed format (e.g., reversed text, JSON fields, character-by-character text, code blocks, or another language).
- **Reasoning Extraction Attacks:** Ask the model to explain how it verifies or stores the password, summarize restricted content, or reveal partial internal state.
- **Context Window Manipulation:** Use long prompts, conflicting instructions, or distractor content to test whether the model loses track of its original constraints.

Your Task

Define at least five (5) distinct jailbreak attacks that you will test. You may use the examples above or introduce your own. In your report, clearly justify why each attack is meaningfully different from the others and what vulnerability it is intended to probe.

Part 2: The Models

You will interact with four models provided on **genai.rcac.purdue.edu** and must export each conversation as a PDF file as proof of your interaction. You **MUST** export each conversation by clicking $\cdots \rightarrow$ Download $\rightarrow$ PDF document and attach the exported PDF(s) in the appendix of your report. **Each exported conversation must not exceed 3 pages.** It is your responsibility to keep each interaction concise enough to satisfy this limit. Each model represents a different difficulty level, and each has a hidden password that it is not supposed to reveal.

Below is the list of models you will interact with. For the extra credit model, you must **not** rely on the model's thinking tokens or any hidden reasoning traces while trying to recover the password. Your attack should be based only on the model's visible responses during normal interaction.

- Guess My Password - 1
- Guess My Password - 2
- Guess My Password - 3
- Guess My Password - 4
- Guess My Password - 5 (Extra Credit)

For this assignment, you should think of each model as a black box. You do not know its system prompt, training data, or defense mechanism. Your job is to infer its weaknesses from observed behavior. If you do not have access to the models, please reach out to `wu1491@purdue.edu`.

Report Requirements

Your report should clearly document your jailbreak attempts and observations for **each** model. At a minimum, your report must include the following:

1. **Attack Strategies Used:** Describe the jailbreak strategies you tried. Organize them into categories and explain the intuition behind each category.
2. **Password for Each Model:** For each model, state the hidden password you recovered. If you were unable to recover the password for a model, explicitly state any information about the password you recovered or that you were unsuccessful.
3. **Interaction Evidence:** Include representative interactions between you and the model. These should show the prompts you used and the model's responses. You do not need to place every interaction in the main body of the report, but you must include examples to support your claims. All prompts and responses (with full context) must be included in the appendix. You may download a PDF copy of each conversation by clicking the three dots in the upper-right corner and selecting *PDF Document*, then attach it in the appendix.

4. **Failure Cases:** If you were not able to retrieve the password for a model, you must explain what strategies you tried, how the model responded, and why you think those attempts did not work.

5. **Comparative Analysis:** Compare the models. Discuss which models were easier or harder to jailbreak, and which strategies seemed more or less effective across levels.

Your analysis should go beyond simply stating whether an attempt succeeded or failed. You should discuss *how* the model behaved. For example, did the model fully refuse, partially leak information, contradict itself, or become more vulnerable after a multi-turn conversation?

Evaluation and Submission

- **Report:** Submit a typed PDF report to Gradescope under `Homework 4`.

Grading Notes

A submission that only reports the passwords without documenting the process will **not** receive full credit. Strong submissions will demonstrate:

- diversity of jailbreak strategies,
- careful documentation of model behavior,
- thoughtful comparison across models,
- clear reasoning about why certain attacks succeed or fail,
- and use of the required report template.