

Project Design and Planning

Ankur Taly (Google)
John Mitchell (Stanford)
Anupam Datta (TruEra/CMU)

Last Three Lectures

- LLMs for Education
 - Personalized tutor, Grading, Teacher assistance
- LLMs and Security
 - LLMs for Security
 - Defense: Fuzzing / Bug finding, Code Analysis, Decompilation
 - Attack: Spear Phishing, Craft code to exploit vulnerabilities
 - Security of LLMs (defending measures used for trust)
 - Break alignment with adversarial prompts
- LLMs for Healthcare
 - Extract structure from EHR, Medical coding, Search and QA

Key steps in any “LLM for X” Project

- **Frame the task**
- **Prompt / Tune the model**
- **Evaluate the model**
- **Assess Reliability and Trustworthiness**

Expanded on next two slides

Key steps in any “LLM for X” Project (1)

- **Frame the task**

- What is the application? What functionality is needed? What are the inputs and outputs?

- **Prompt / Tune the model**

- Design a prompt, Include few-shot examples
- Fine-tune weights (if you have training data)
- Retrieval Augmented Generation (RAG) for supplying relevant context

- **Evaluate the model**

- Gather evaluation datasets
- Define a set of metrics. Tricky for generative tasks.
 - There may not be a canonical label
 - Think about what would make a good response

Key steps in any “LLM for X” Project (2)

- **Assess Reliability and Trustworthiness**

- **[Grounding]** Ensure that responses are always grounded in some knowledge source and not “made up”
- **[Confidence]** Quantify uncertainty / confidence of responses
- **[Interpretability]** Understand how / why the model generated a response?
- **[Robustness]** Assess the robustness of the model to adversarial prompts

Example: Better Homework Grading (1)

- **Frame the task**

- What is the application? - A tool for grading homework or exam questions, given rubric
- What functionality is needed? - Accurately score solutions and **give useful comments**
- What are the inputs and outputs? -
 - Input: Question, solutions to evaluate, grading instructions / rubric
 - Output: Score + Explanation / Comments?

- **Prompt / Tune the model**

- Let's assume there is existing work on accurate scoring; try to generate useful comments
- Build on LLM explanation of program errors to see if LLM can explain homework errors

Example: Better Homework Grading (2)

- **Evaluate the model**

- Gather evaluation datasets - There may be existing grade data from Stanford CS class
- Define a set of metrics. Tricky for generative tasks.
 - There is controversial work from MIT on using LLM for grading exams
 - Possibly use LLM to evaluate explanations?

Example: Better Homework Grading (3)

- **Assess Reliability and Trustworthiness**

- **Interpretability**

- Why this score? Why this explanation?
- Is the explanation for the score faithful to the solution and instructions?
 - E.g., Explanation says “solution does not mention X” when:
 - it actually does OR
 - X is irrelevant to the problem at hand

- **Confidence**

- How certain is the model of its score?

- **Robustness**

- Could a student cheat by slightly tweaking their incorrect solution and have it be accepted by the LLM as correct?

Example: Summarize doctor notes (1)

- **Frame the task**

- What is the application? - A tool for summarizing doctor's notes
- What functionality is needed? - Comprehensively and faithfully summarize a doctor's note
- What are the inputs and outputs? -
 - Input: Free text doctor's note
 - Output: A table extracting various dimensions (prescription, diagnosis, labs ordered) from the note

- **Prompt / Tune the model**

- Write a prompt describing the task
- Include a few examples in the prompt so that the model understand the expected form of output

Example: Summarize doctor notes (2)

- **Evaluate the model**

- Gather evaluation datasets - There may be existing structured summarization datasets in other domains (e.g., retail product descriptions -> feature spec)
- Define a set of metrics.
 - Precision, Recall of identified features

Example: Summarize doctor notes (3)

- **Assess Reliability and Trustworthiness**

- **Grounding**

- Every identified feature in the summary must be present in the note
 - Recall 'diabetes' example from Divya's talk

- **Interpretability**

- Why this feature? What part of the note is this feature-value pair coming from?

- **Robustness**

- How robust is the model to seemingly benign perturbations to the note, e.g., add / drop punctuations, add / drop stop words
- How robust is the model to order of information in the note, for instance, what if prescriptions are mentioned before the diagnosis?

More about reliability and trustworthiness

- **Grounding**
- **Confidence**
- **Interpretability**
- **Robustness**

Expanded on next four slides

Grounding

An LLM response is considered **grounded** if **every claim in the response can be attributed to an authoritative knowledge source**

Two parts:

- Improving grounded-ness of responses:
 - Retrieval Augmented Generation, Prompt model to not use information beyond the context,
...
- Verifying ground-ness of responses:
 - Use an Natural Language Inference (NLI) model to compare response (hypothesis) to context (premise)
 - Read: TRUE: Re-evaluating Factual Consistency Evaluation
 - Sample multiple responses, and check consistency
 - Read: SelfCheckGPT: Zero-Resource Black-Box Hallucination Detection for Generative Large Language Models

Confidence

Establish a **level of confidence / certainty** for LLM responses

- Quantify it using a calibrated numerical probability score
 - Read:
 - [Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation](#)
 - [Just Ask for Calibration: Strategies for Eliciting Calibrated Confidence Scores from Language Models Fine-Tuned with Human Feedback](#)
 - [Teaching Models to Express Their Uncertainty in Words](#)
- Incorporate uncertainty in the response text, "***I am not sure but the answer may be: ...***"
 - Read: [Reducing conversational agents' overconfidence through linguistic calibration](#)

Interpretability

Understand/Explain/Interpret **how** the model came up with the response

Tracing a response to parts of the prompts

- Perturbation / Ablation, Shapley values
 - Read
 - [The Explanation Game: Explaining Machine Learning Models Using Shapley Values](#)
 - [Anchors: High-Precision Model-Agnostic Explanations](#)
- Examine gradients to inputs tokens
 - Read: [Axiomatic Attribution for Deep Networks](#)

Tracing a response to training / fine-tuning set

- Influence functions, TraIn (requires access to training checkpoints)
 - Read:
 - [Understanding Black-box Predictions via Influence Functions](#)
 - [Studying Large Language Model Generalization with Influence Functions](#)

Robustness

Examine if the model is robust to **adversarial inputs**

Design adversarial inputs that fool the model

- Making the model return an incorrect outcome
 - Read:
 - [Semantically Equivalent Adversarial Rules for Debugging NLP Models](#)
 - [Beyond Accuracy: Behavioral Testing of NLP Models with CheckList](#)
- Make the model generate bad (racist, abusive, inappropriate) text
 - Read: [Exploiting Programmatic Behavior of LLMs: Dual-Use Through Standard Security Attacks](#)

Design mitigations that guard against adversarial inputs

- Input filters, Output filters, Model tuning (often using human feedback; RLHF)

Sample education project directions

- **Clara - writing assistant for Google docs**
- **Bruno - conversation to transcript tool for LLM analysis**
- **Coding error message explanations**
- **Contrasting cases**
 - Designed as teacher tool
 - Would it work for students?

More details in later slides

Invite **Clara AI** for formative writing feedback

Currently in limited research beta testing

Add **clara@uphold.ai** as an editor to your document then submit the URL

<https://docs.google.com/document/d/82TY...>

Invite →



Ben Klieger



Record

Transcript

Say something to begin.

Instructions

1. Start talking about your project.
2. When you feel stuck or unsure what to do next, ask the following prompt: "Who spoke the most in the conversation?"

Prompt + Response

B

Bruno AI

Bruno AI's feedback will appear here. This is a placeholder to show where the feedback will show.

U

You


This is a past message made by the user to Bruno.

B

Bruno AI

This is Bruno's response.

Send a prompt

 [Reset Session](#)

Ask Bruno

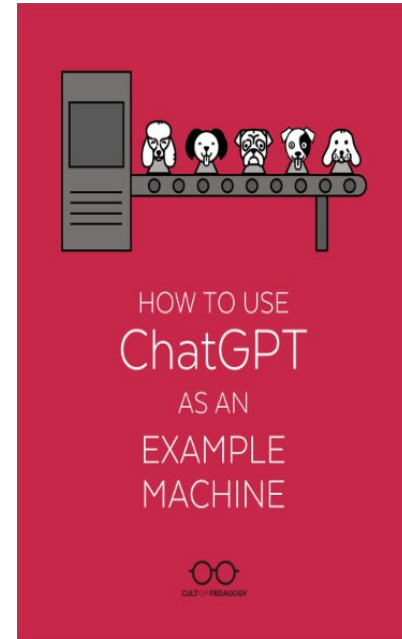
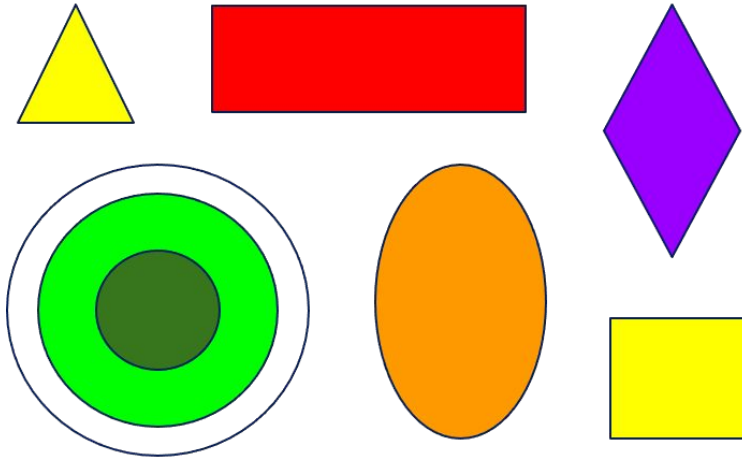


Enhanced error messages with GPT

- Compare two approaches to baseline options
 - Generate explanatory error messages using OpenAI's GPT in real time
 - Construct error messages that link to the course discussion forum
- Result
 - Students using GPT-generated error messages
 - Repeat an error 23.5% less often in the subsequent attempt
 - Resolve an error in 36.1% fewer additional attempts, compared to standard error messages

AI-generated teaching examples [Mah, Levine]

- Contrasting cases



- A general concept is best illustrated using two or more contrasting examples

Additional education project ideas

- lesson planning - TeachAssist (Riz Malik)
- neurodiversity - ADHD FlexABLE ai <https://ed.stanford.edu/ldt/students/projects/flexibleai>; neurodiversity and creativity in Bangladesh (Labib Rahman)
- language learning - KATE (Ted Song)
<https://ed.stanford.edu/ldt/students/projects/kate-knowledgeable-ai-tutoring-english>
- learners as teachers - study buddy (Olivia Tomaneo) <https://ed.stanford.edu/ldt/students/projects/studybuddy>
- personalized early reading: Ello (CS grad working with Nick Haber)
https://www.helloello.com/lps/reading-confidence?utm_source=bing&utm_medium=cpc&utm_campaign=B_S_Brand&utm_term=ello&msclkid=9316c8ce6f2319a326efadd7c66c8de6 and Project Read (Ramakrishnan, GSB)
(<https://www.gsb.stanford.edu/experience/news-history/vivek-ramakrishnan-mba-23-how-ai-could-help-solve-school-literacy-crisis>)
- intersection of VR/AI for engagement - Alex Stolyarik [we are launching an interesting program in that area with Unity, led by Kristen Blair]
- AI literacy - CRAFT - Victor Lee, Parth Sarin <https://craft.stanford.edu/>
- Novice approach to programming - Benjamin Xie <https://www.benjixie.com/publication/icer-2023/icer-2023.pdf>

Sample security project directions

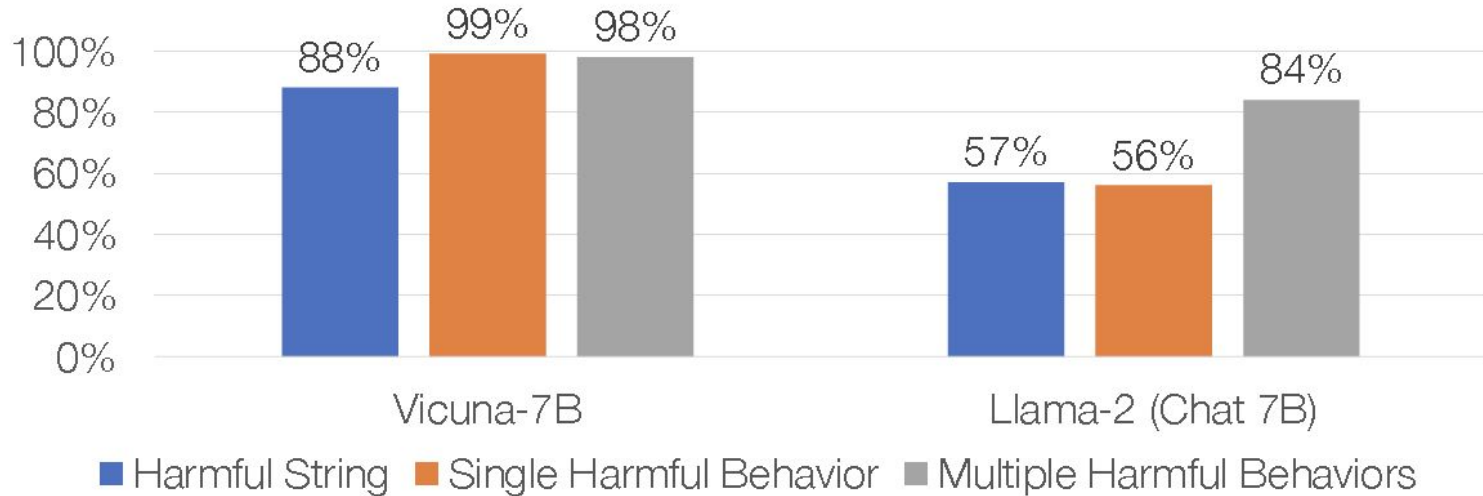
- **LLMs for Fuzzing / Bug finding**
- **LLMs for Attacking other models, software (or people)**
 - Spear phishing, Craft code to exploit vulnerabilities
- **Assess LLM Robustness**
 - Attack: Techniques for breaking alignment
 - Defenses: Techniques for detecting adversarial prompts
- **Detecting whether a text is LLM generated**
 - Techniques: Watermarking, Examine structure of an LLM's probability function
 - Read:
 - [A Watermark for Large Language Models](#)
 - [DetectGPT: Zero-Shot Machine-Generated Text Detection using Probability Curvature](#)
 - Attacks: Techniques for defeating detectors
 - Read: [Can AI-Generated Text be Reliably Detected?](#) (more details ahead)

Mitigating Security Risks (Workshop)

- GenAI technologies have changed the computing landscape
 - Enabled exciting applications,
 - Help adversaries generate spearfishing emails or spread misinformation
- Workshop on the risks of GenAI considers
 - How could attackers leverage GenAI technologies?
 - How should security measures change in response to GenAI technologies?
 - What are some current and emerging technologies for designing countermeasures?
 -
- The [June workshop](#) was organized by Google, Stanford, and UW-Madison
 - Second Oct 16 coming up; stay tuned for new ideas

Security of LLMs

Attack Success Rate on Open Source Models



How do we fix this?



- **Open direction for future research**
 - Experts have been trying to fix adversarial examples in computer vision for ten years
- **Room for experimentation**
 - Repeat attacks based on open release of experimental methods
 - Compare possible defenses
 - Look for new ideas

Sample healthcare project directions

- **Extract structured data from narrative text**
 - E.g., billing codes from notes; symptoms from narratives, plans from doctors' and nurses' notes
- **Generate narrative text from structured data**
- **Generate reports from non-narratives (imaging, signals, ...)**
- **Question answering**

More details in later slides

Healthcare ideas from LayerHeath

- Extract structured data from narrative text
 - E.g., billing codes from notes; symptoms from narratives, plans from doctors' and nurses' notes
- Generate narrative text from structured data (??!!)
- Generate reports from non-narratives (imaging, signals, ...)
- Combine text models with other data to create models for cohort selection, outcome prediction, ...
- Summarize vast numbers of notes to what's important. (For what use cases?)
- Question answering

Dataset suggestions from LayerHealth

Dataset	Description
Clinical Trial Matching	All FDA clinical trial eligibility criteria are freely available online.
Medical Information Mart for Intensive Care (MIMIC)	Vast dataset of de-identified structured & unstructured clinical data across ICU and ED.
PMC Patients	Patient summaries extracted from PubMed case reports; 167k+ patients.
Adverse Drug Event Corpus	Extracts all adverse drug events (ADEs) from a set of clinical notes.
Synthetic note generation	As in here , generate synthetic notes

Student interests

- Aman Kansal
 - MScS student in AI, interested in application of LLMs in healthcare, edtech, and security
- Pooja Sethi
 - MS CS student in the HCP program, interested in the intersection of vision + LLMs as well as multilingual models; based remotely in Seattle, happy to hop on video calls
- Kevin Marx
 - 2nd year EE master's student focusing in hardware and bioinstrumentation, interested in medical diagnostics in resource constrained environments.
- Jerry Shan
 - MS in Learning Design and Technology at the GSE, interested in applications of LLM in edtech, particularly related to computer science education
- Aryan Siddiqui
 - sophomore deeply involved in the LLM space (worked at Kick developing an agent to integrate with financial platforms...), interested in security and education

Thoughts forward: Jerry and Matt

- Use LLM to build an AI-assisted code editor to guide computer science students solve coding questions without spoiling the solutions.
- Contracts (construction/law) typically are long and difficult to locate exact pieces of information. This project can focus on building a question-answering system using LLMs plus retrieval based off supplied documents.

Thoughts forward: Dora and Andrew

- To promote transparency in ML dataset collection, one popular proposed method is creating datasheets (Geburu et al.). We propose leveraging LLMs that are grounded in the existing research documents dataset ... to generate datasheets.
- There is a disproportionate amount of healthcare burden placed on certain patient groups (e.g., Black women). We want to empower patient advocacy by using LLMs to help them practice conversations with their clinicians.
- Research papers in computer science and machine learning often describe new algorithms that we may want to implement ourselves. We can use LLMs to ingest the contents of research papers and produce code in a language of our choice.

DISCUSSION