CIS 7000-005: Debugging Data & Models Fall 2022

Project Proposal — September 6

Prof. Eric Wong

The course project is your chance to pursue a research problem in the area of debuggable machine learning. Generally speaking, the project should involve the inspection and debugging of a machine learning problem. All parts of the pipeline are fair game, including data collection, training algorithms, models and architectures, the resulting predictions, and even the debugging tools themselves. This can take the form of an audit (identifying the shortcomings of a fixed pipeline) or a patch/update (changing the pipeline to fix a problem). The specific setting and application is ultimately up to you.

1 Structure of the proposal

Your proposal should generally have the following sections:

1. Introduction - Introduce the setting and motivate the problem for a *broad ML audience.* Assume that your reader has an introductory ML background, and is not necessarily knowledgeable about your specific setting. What is the big picture problem that your proposal takes place in? Why is this important? How have people in this area tried to tackle this problem? What are the benefits and drawbacks of existing approaches, and how would your proposed work fit in here?
2. Related work - Survey the related literature in more detail at the level that a *researcher familiar with the area* would understand. For example, what related approaches have people already tried in this space? What work does your proposal build upon?
3. Proposed work - Describe the work that you want to do in the course project, as if you were describing the project to *one of your fellow students.* Again assume that your reader has an introductory ML background but not necessarily any specific knowledge. Be sure to describe at a conceptual level what you are trying to do, what you hope to achieve, and why you think the approach may work.

1.1 Style guidelines

Use the NeurIPS stylesheet at `https://nips.cc/Conferences/2020/PaperInformation/StyleFiles`. The proposal should be at most 2 pages excluding references.

2 Logistics

Logistics for groups and presentations will be handled at the following spreadsheet:

https://docs.google.com/spreadsheets/d/1sjSnNAXARiqjTDlXRrP-lAyutgTvJ_fk36bOOBnJ3ZO/edit?usp=sharing

2.1 Groups

Learning from your colleagues is one of your best resources. People with different backgrounds bring different perspectives, and helps you understand how your research appears to other researchers. You are *strongly encouraged* to form groups of 2 to 3 people. If for some reason you need to work in a group by yourself, please check in with me before doing so.

To help facilitate group finding, once you have a course project that you would like to pursue, go to the *Project Groups* and enter a brief description (1-2 sentences) of your project as well as the names and emails of anyone in your group. If you do not have a full group or wish to join another group, you can reach out to any other incomplete groups on this sheet. You can also post on Ed Discussion.

2.2 Sign up for Project Checkpoint Presentation

Once you have your group, go to the *Sign-Up* sheet at and sign up for a Project Checkpoint presentation. There are slots for two projects per day between October 11 and November 17th.

3 Example research questions

Example questions that can lead to a research project at various stages in the pipeline include the following:

Datasets.

- Are there biases, spurious correlations, or underrepresented subpopulations? For example, in the case of US census data, what are the blind spots or misleading correlations that ML models learn?
- What is the underlying source from which these problems stem from? Can you quantify or pinpoint the impact downstream predictions?
- Can we fix the data or collection procedure to mitigate these issues? What would your recommendations be to policymakers or engineers?

Methods and architectures.

- Do ML algorithms (i.e. fairness / privacy / adversarial robustness / security) for fixing models via training actually achieve their goal? When do they miss the mark?
- Can you pinpoint or characterize the failures of modern architectures (such as in language or vision models)? Are these systemic issues?

- Can you construct counterexamples / subpopulations that exemplify the failure modes of these models and algorithms, or guarantee that such failure modes don't exist?

Interpretability and predictions

- How faithful is a particular explainability method to the actual model predictions? Is this a general trend or application-specific behavior?
- Are the type of explanations we can generate aligned with what practitioners need? What information do practictioners need anyway?
- For example, do analysis tools for diagnosing health conditions tell doctors useful and meaningful information? What do doctors need, what can existing approaches do, and how do we bridge this gap?