SI 650 / EECS 549: Project Update

Version 1.0
Due: **Friday, November 18**, 11:59pm

1 Introduction

The course project is intended to provide an opportunity for students to dive deeper into one problem or topic of their choice and write a very small scale study on the topic. Ideally, your course project is a chance to develop something you can talk about during job interviews to show deep experience in a topic, code you can show off to future employers to demonstrate technical expertise, or the results could serve as a pilot study for a full research project.

2 What to do (10 points total)

The project update serves two purposes. The first is that it helps you write an initial rough draft of your final project report. The second is that it requires you to have all of your data in hand and to develop a simple baseline for your proposed task. Both of these are important steps at keeping your project on track. Following, we outline the key requirements for your update.

Introduction (1 point) You should have a rough draft of the introduction that clearly states what the problem is and provides some broader context. We recommend writing the introduction last after you finish the Problem Definition and Related Works sections. You should also include a statement on why solving this problem matters–who would care if you solved it and what effect would solving it have? In the project update, you should include details of your proposed method for solving the problem (even if you haven't implemented it).

Data (3 points) The section describes what data you will use for the project. For the update, you should already have the data on hand. You should describe the source of the data, how you obtained it, what type of preprocessing steps you took, and (if not sensitive data) include a few examples. We strongly encourage you to include some very rough statistics (e.g., how many instances you have, the class distribution if doing classification, relevance score distribution) in a table format. If you had to create your own dataset or needed to annotate a ground truth relevance, this section should specify how you did it and provide details on the relevance scores.

If your data doesn't have the required ratings to evaluate, e.g., you're building an IR system but the data doesn't have query-document pairs scored by relevance, you will need to annotate. The amount of annotation depends on the task:

- For IR systems where you enter a query (most projects), each person in the project should annotate *at least* 20 queries and a large selection of documents for each of those. We typically ask you to provide relevance scores for 100-200 documents per query. You should probably not rate documents at random! Instead, a common strategy is to use a variety of approaches to retrieve documents (e.g., rate the top 50 BM25-retrieved documents, top 50 dense-vector retrieved documents, etc...) and to also pick some query-related keywords you, as the expert, know to be related and find and score documents with those. We typically recommend using a 5-point relevance scale for rating (from not-relevant to highly-relevant).

- For classification-based projects, we typically ask for 500-1000 annotated items per project member. Some of these will likely serve as training data ($\sim$80%), while the rest will be test and development data.

- For recommender systems where you are estimating user-item scores, you will ideally want real data (not annotated data). However, if you don't have this, you should create different personas and rate items according to each persona's preferences. We also encourage you to discuss this with the instructors prior to starting.

Getting a good quality dataset for evaluation is one of the most important parts of the project and is the bulk of the work you will do for the project update. Creating this data is often a good way to understand what is going well (or wrong) in your IR systems, which can help you build your final model better. Please see us early for questions or post on Pizza!

For the data you create, **please create a diverse set of queries that show off the different abilities of the models**. For example, not all queries should be single-word queries. In vertical IR domains, people may want to search for complex things; for example, in a recipe search engine you might want to ideally support a query like "spicy chicken recipes that take under 30 minutes and don't need an oven". It's totally fine if your system doesn't do well on these. Your job as data creator is to think about what an *end-user* might want to do with your system.

Related Work (1 point) The related work section should describe how other people have thought about the problem you're working on. How did they approach it? What makes their problem different from yours? Why do you think your approach will be better? For your update, you should have at least five papers related to your current problem and a few sentences describing what they did to solve the problem. We recommend using Semantic Scholar or Google Scholar to help find related papers.

Methodology (1 point) This section will describe how you solve your problem. Go into algorithmic details and be sure to describe what various kinds of preprocessing steps you did. Someone should be able to recreate your exact methodology from the description. Be specific about what each step does. For example, it's insufficient to say "we trained a classifier;" instead say something like "we trained a Random Forest classifier using 250 trees and requiring a minimum of 5 items per leaf" For the update, you should include a *detailed* outline of the method you plan to try, even if you haven't implemented yet. Your update describe to a reader what you want to do and *why* you want to do it. Think of this part as an exercise in writing a full description of how you plan to solve the problem. This update also lets us give you feedback on different parts of your plan.

Evaluation and Results (3 points) This section provides an overview of how you evaluated your method on the data. What methods did you compare against? How successful were you? Describe the exact evaluation setup and what kinds of steps were taken.

For the update, you should clearly define one or more baselines to compare your system against. One baseline should be random performance. A second baseline should be something reasonable that doesn't require much knowledge or learning. For example, if you're doing a classification, always choosing the most frequent class is a useful baseline; or if building a search engine, evaluating against BM25 or tf-idf for retrieval. *Generating a result on your actual data with a real baseline method is the most important part of the update and will have the biggest effect on your grade.* You need to demonstrate that you can work with the data to solve the problem (even poorly with a baseline method!) so that when you actually try to solve it with your own method, you know how to work with the data and know how to evaluate your system. If you're having trouble coming up with a baseline, please see one of the instructors immediately. If you don't have data for the problem you're working on, consider switching to a different task where you can get the data.

You should have at least one figure or table showing your baseline's results. Please make sure to label all your axes and make the font size legible without having to zoom in excessively.

Work Plan (1 point) For the update, describe your workplan for the semester in terms of (i) what you've done so far and (ii) what you intend to do to finish the project. Be specific and lay out weekly objectives/milestones that you can use to keep track of progress. We won't hold you to this workplan for the final project, but we have found that the act of creating such a workplan generally helps students quantify the remaining work involved and scope their project better (fewer December surprises!).

What to submit?

You need to submit one thing due by the deadline:

1. Upload a PDF of your report to Canvas.

Everything needs to be submitted to Canvas.

Late Policy

Throughout the semester, you have three free late days total. These are counted as whole days, so 1 minute past deadline result sin 1 late day used (Canvas tracks this so it's easier/fair). However, if you have known issues (interviews, conference, etc.) let us know at least 24 hours in advance and we can work something out. **Special Covid Times™ Policy**: If you are dealing with Big Life Stuff®, let the instructor know and we'll figure out a path forward (family/health should take priority over this course). Once the late days are used up, the homework cannot be submitted for a second, though speak with the instructor if you think this is actually a possibility before actually not submitting.

Academic Honesty Policy

Unless otherwise specified in an assignment all submitted work must be your own, original work. Any excerpts, statements, or phrases from the work of others must be clearly identified as a quotation, and a proper citation provided. Any violation of the University's policies on Academic and Professional Integrity may result in serious penalties, which might range from failing an assignment, to failing a course, to being expelled from the program. Violations of academic and professional integrity will be reported to Student Affairs. Consequences impacting assignment or course grades are determined by the faculty instructor; additional sanctions may be imposed.