

SI 650 / EECS 549: Final Project Report

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 show off to future employers or could serve as a pilot study for a full research project.

2 What to do (15)

The final project report serves three purposes. First, the report should **summarize all of the things that were done** as a part of completing your project. Not all of these things will have worked, nor will all of the things that worked have worked well, but that's still fine. The goal of the project overall is to have you explore and go in-depth into a particular topic, so the report is your chance to **describe your efforts along these fronts**. Second, the final project report is an exercise in **describing technical material to a technical audience** (this is in contrast to the blog post, which is for a general audience). Often times, these types of reports will be for internal distribution to managers, colleagues, or other practitioners in the field. In this report, you should demonstrate your **communication skills**. Third, the project is a chance to show off what you learned as a part of the project. Some of what was learned will come from insight from the project itself and learning what works well at solving a particular problem. However, some of the insight can come from what you yourself learned about the development process, the task itself, project planning, etc.

Below, we have summarized the main sections that should be in place for your final report and what content is needed for each (along with their point values). If you have questions on anything, please let us know on Piazza!

Introduction (1 point) Your final report should have a polished introduction that briefly summarizes (1) **what your project is doing**, using plain language, (2) **what others have done and why your approach is different**, (3) **what you did and how well it worked**, and (4) a statement on **why solving this problem matters—who would care if you solved it and what effect would solving it have**—and (5) what are the **main contributions from your project** (e.g., what was learned? what's the one thing you would tell someone so they could get insight from your hard work). The draft of your Introduction from the Project Update likely contains much of this text, so you can just revise/style the text to make sure these five points are covered.

Data (0 points)—but still needed The section describes what data you actually used for the project. This section should retain most of what you had in the update, but if you changed anything, you should include additional or revised text describing any differences from what you had in the update versus what you used for your final report. 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. You should be clear about the statistics for your final dataset (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.

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 final report, 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. In the final report, please revise or add to your descriptions of these related works with respect to what you ended up doing or what results you found.

Your related papers don’t need to be exactly on your specific problems but can be on related problems (where you learned something) or on more general kinds of specialized search that your project is an example of.

Methods (5 point) The final report should contain a detailed description of what methods you did to solve your problem. Walk the reader through details such as how you processed your data, what features you used, or which methods you used. Cite any references if you are using an established technique. If you are using a library, list the libraries you used. Ideally, your description should help the reader understand why you made the design decisions you did (e.g., “why these features?”). Note that if you tried some things that didn’t work out, you can describe them later in the final report too in the “Other things we tried” section.

The course project is intended to help you grow beyond the skills you learned in class. For full credit, your final project should demonstrate that you have learned new techniques and methods. For example, simply re-using most of the code from class assignments for use in your project would not receive full credit.

Evaluation and Results (4 point) 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 final report, you should have evaluated your baselines and your proposed model (described in the Methods section). Remember, your system doesn’t have to work well; it just needs to work. If you’re having trouble getting some result, consider simplifying your system until you can get something working. Then you can build on top of that working system (if you want) to improve model performance.

For the final project, it’s often a nice touch to report on the effects of different parts of your system on performance, e.g., hyperparameters, features, heuristics. For example, if you’re running a classifier and have 100 features that could be grouped into different categories, what’s the effect of leaving out certain kinds of features? This is what’s known as an ablation study. Similarly, if you

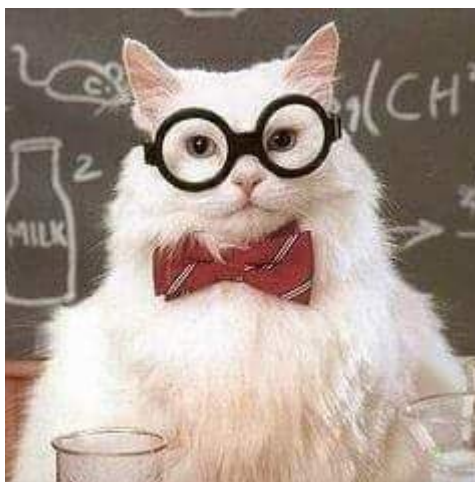


Figure 1: Feel free to include a figure. Make sure your axes are labeled and the font is big enough. You can re-use these for your blog post too! Seaborn is a great plotting library to make figures in.

have a **hyperparameter like vector size**, what's the effect of changing this on the final performance (You did this in HW1 when you set the smoothing rates for Naive Bayes!).

You should have **at least one figure or table showing your baselines' and model's results**. Please make sure to label all your axes and make the font size legible without having to zoom in excessively. You can **discuss the figure and/or table next in the Discussion section**, as they help the reader understand what the model is doing.

Discussion (2 points) The discussion section is where you start to unpack the results for the reader to help them understand what was learned. You can discuss here **what has gone wrong and right** in your current setup. For example, maybe you realized you needed more data, or maybe you realized that the SVD was not helping your analysis. The discussion should point the way to what work will be done next.

In the final report, you should discuss **how well your baselines and model did and what findings you can draw from this**. The key part to a discussion is **not stating the results** (e.g., “the baseline got a 12.0 NDCG.”—this is bad; don't write that) but rather trying to *explain* the results to the reader and help them understand **why these results matter** and how to interpret and contextualize the result. For example, if your method's performance is low, why is it low? Is there something about the data? Or, **if the performance is high, is this expected and why?** It's useful to discuss where your model performs well and where it doesn't. If you look at the kinds of mistakes it makes, are there any patterns? What was the effect of changing the model parameters or holding out certain features?

In the final report, discuss (1) how well your approach did overall—is this performance satisfactory for an end-user? is it enough to do useful science? (2) how well did your approach do with respect to the baseline? Was the performance closer than you expected? What might explain the difference in performance? and (3) Why your approach did well (or badly—it happens). Do *not* just report numbers—put your numbers in context and discuss **why these numbers matter** and what the reader should learn from seeing the performance numbers.

Conclusion (0 points—but still needed) Wrap up the main part of your paper here with a conclusion that summarizes what you did and what you think are the main conclusions. This is your chance to remind the reader of what all you have accomplished and re-tell the story of your paper. If you think there are interesting future directions that a reader might pursue, describe them here. If you put your code on github or some other open source repo, link to it here as well.

Other Things We Tried (1 point) We fully realize that not everything will go according to plan during the projects! That's a part of data science and using (or developing) NLP techniques for real world data. You can use this section to describe things you tried but couldn't get working. If you spent a lot of time on some aspect that's not included in the previous sections, describe it here. This section is here to help convey your effort so you can get credit for being ambitious. Be sure to describe things in detail though (e.g., saying "we spent 30 hours prototyping a deep learning system but it didn't work" doesn't tell us much so it's tough to provide credit for this.

What You Would Have Done Differently or Next (1 points) Projects are fun but can be frustrating since you're working on a problem that doesn't have a fully-specified solution yet. If you had to **start the project over**, what would you have done differently? Are there ideas you would have wanted to try but didn't have time? Did your results point the way to some next step that you think might work? Use this section as an opportunity for reflection on how the project went.

What to submit?

You need to submit one thing due by the deadline specified on Canvas:

1. Upload a PDF of your report to Canvas.
2. The code you used, ideally as a zip file, with a **README on how to run your IR system** interactively.

Everything needs to be submitted to Canvas.

Late Policy

By the time the final report is due, the instructors need to turn in grades very soon, so **no late days can be used on the final report**. However, if you have known issues (interviews, conference, etc.) let us know at least 24 hours in advance and we can work something out.

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. This includes example code provided from worked solutions (e.g., something on Kaggle or in a github repository), but does *not* include the kinds of snippets you might see on StackOverflow.

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.