CMPS 242 Project Evaluation

This document contains important details regarding your project submission and evaluation. It is important for each member in your team to be aware of these.

Grading Rubric

We will be using the following grading rubric for your final project submission.

Criteria	Grade %	Detailed Comments of what we expect
Novelty in Problem Formulation	15 %	How original or unique is your use-case, problem compared to the ones you learnt in class?
Novelty in Approaches	15 %	Are you trying out something different / unique than what was already covered in the class/homeworks?
Data Cleanup / Feature Engineering	10 %	Did you use the entire data? If not, did you sub-sample a reasonable amount? Are the features adequate and reasonable? What pre-processing did you do?
Implementation	20 %	Is your algorithm implemented correctly and from scratch? Is code repository well organized, contains unit tests and readable?
Evaluation / Baselines	20 %	Do you have some metrics to measure performance of your algorithm? Or, any existing baselines you compare against?
Key Results / Observations	20 %	Do you have interesting findings from your project? What are they? Lessons learnt? Things that went wrong? Challenges?
TOTAL	100 %	

To put things in perspective for you, we have also gone through your mid-term project reports carefully keeping the above criteria in mind and have emailed you our detailed feedback. This should give you an idea on which areas you need to focus on for the next two weeks. If you have questions, you can come by during our office hours.

FAQs:

- **What is the format expected for the final project submission?**

Please read this carefully. Points will be automatically deducted if your submission does not obey these guidelines.

The final report should be 4 pages (front and back, NOT single sided) in ICML template format. On the first page, include your Project Title at the top, followed by team members,email-ids and link to your code repository.

In the main body of your report, include the following sections (mandatory): Problem Statement, Feature Engineering, Algorithm/Model Formulation, Evaluation, Results, References (if any). You may add more sections if you want to.

We are going to be accepting electronic copies for the final report. Your final report should be a file named as: *CMPS242_proj_<teamname>.pdf* and emailed to both the TAs. The subject line of the EMAIL should read

[CMPS242] We're done with this - team #

where the team number is the number that you got when you received the midterm report.

- **How is our project submission different from our homeworks?**

Even though part of the intention behind HW3 was to help you get some implementations that you could re-use for your project, please bear in mind that the there has to be some value addition in your final project reports. Your project report has to be substantially more detailed than what you would submit for a homework. For instance, you cannot merely report the accuracies with few methods you used and add the equations. We want to see how you are going to use them to solve the task on the yelp dataset. So, discuss the possible reasons behind your results, what do you want to try next to improve them, are there scalability challenges if you pursue a particular method?

- **What do you mean by novelty?**

We want to look at what your project brings to the table in addition to the material that was covered in the lectures, homeworks and sections. Therefore, if your problem formulation or use-case is not very novel, you can compensate for it by coming up with a novel approach/algorithm (that was not covered in the class/hws). Similarly, if you use existing algorithms from lectures/hws, then you you can bring in novelty in the problem formulation or use-case you are addressing.

- **Are unit-tests mandatory?**

YAAAAAASSSSSSS!!!!!! Yes, we have a small percentage of grades for them. This is an opportunity for you to get exposed to testing a piece of functionality in your code.

- **Will you be looking at our source code?**

YOU BET! Yes, we will be reviewing it. Please provide a link to your repo in your final report.

- **How do we list the work split across the members of my team?**

This is important. Include a text file contribution.txt mentioning the contribution of each of your team members. We might ask either all or one of you to explain your project (and code) to us, so please make sure everyone in your team is familiar with the project.

- **What do you mean by code being well organized?**

Glad you asked! Your repository should separate different elements of your project so that it makes it easier for others to read your code, collaborate etc. For instance, make source, test, data directories. Put in a README at the top level directory summarizing the aim of your project and instructions on how to run your code. It is NOT OK to have one long ipython notebook file, please separate them out into source files. If you want to include results, plots, documents etc too you can create additional directories for them.

- **How important are results? Do accuracy numbers matter a lot?**

The results section of your project should highlight the insights you gained from your project. We do not strictly look for high accuracies as a grading criteria, so it is ok to have obtained lower numbers as long as you give meaningful justifications for them. If something failed, mention why you think so and what you could do to possibly fix it.