**Final Project Report for CS 184A/284A, Fall 2019**

**Project Title:**  title goes here  
**Project Number:** number goes here

**Student Name(s)**Name1, StudentID1, uci\_email\_address(Name2, StudentID2, uci\_email\_address)

**General Instructions:**

* *Your report should be 6 to 8 pages long in PDF*
* *If you want to add more details (e.g., additional graphs, examples of your system’s output, etc) beyond the 8-page limit, feel free to add an Appendix to your report for additional results*
* *The sections where I expect to see mostly new material, and where I will focus on the most for grading, are Sections 4, 5 and 6, since these are the sections where you should have the most new material to report relative to earlier reports. Don’t ignore the other sections, but pay particular attention to Sections 4, 5, and 6.*
* ***What the Team submits to Canvas (one student submits the items below on behalf of the team)***
  + *Your report entitled FinalReport.pdf*
  + *A zip file called project.zip that contains the following in 1 directory called project/*
    - *A README file that contains a 1 line description of each file in project/*
    - *A Jupyter notebook (called project.ipynb) that can be run directly and that demonstrates your project. Your notebook can import a sample of the data that you used, import 1 or more models that you built, and generate examples of the types of predictions or simulations your model can make. The notebook should not take any longer than 1 minute to run in total (if you have models that require a lot of training time, train them offline and just upload the models and some sample data to illustrate them). Feel free to generate examples of your model(s) in action, e.g., for reviews you could generate examples of reviews where the models work well and reviews where the models work poorly.*
    - *Also save a .html version of your notebook called project.html, showing the outputs of all the cells in the notebook.*
    - *Upload any data files needed to run project.ipynb – keep your data sets to 5MB in total or or less.*
    - *Also include a subdirectory called src (within the zipped project/ directory) with all of the individual code (scripts, modules) for Python (or equivalent for other languages) that your team wrote or adapted– these don’t need to be called by the project.ipynb notebook but need to be in the src/ directory*
    - *Note that we don’t necessarily plan to run all your code, but may want to look and run parts of it.*

**1. Introduction and Problem Statement (1 or 2 paragraphs)   
*[This can be similar to what you wrote in your proposal or progress report]****A brief summary that summarizes your project and your main results (essentially like an abstract for a technical paper).**Define precisely what problem your project addressed. For example if your project is multi-label document classification then you would clearly define what multi-label document classification is, any assumptions you are making in your problem setup. Write this section so that anyone with a degree in computer science could understand clearly what you are talking about.*

**2. Related Work: (1 or 2 paragraphs)***Write 1 to 2 paragraphs describing what methods/algorithms have been used in the past to address this problem. Provide a few references to research papers or articles that describe previous work on this problem. Describe how your project fits in the context of earlier work, e.g., “we systematically evaluated the performance of standard methods (as described in X, Y, and Z) on several data sets, rather than developing new algorithms.”*

**3. Data Sets [at least 1 page]  
*[This should have considerable detail – make sure you include a good description of your data set(s) – figures and tables are strongly encouraged. Can be an updated version of what you wrote before.]****Describe what data set(s) you used in the project – include references (e.g., URLs) for where you obtained the data if you can. Feel free to include figures in this section.*

**4. Description of Technical Approach [at least 1 page]  
*[Can be an updated version of what you have written in earlier submissions….]****Provide a description of the techniques and algorithms that made up the core of your project. For example, if your project involved comparing different classification algorithms for document classification then in this section you would list and briefly describe the classification algorithms you used (e.g., naïve Bayes, logistic regression, support-vector machines, recurrent networks, etc). Be as clear as you can about what versions of algorithms you used.*

*You can include descriptions of preprocessing software, API/crawling code, data cleaning scripts, etc. Also feel free to include descriptions of algorithms and software you spent time on developing but that did not end up being part of your final project for one reason or another.*

*It may be helpful to show a block diagram that shows how the different pieces of your system work together, e.g., a pipeline of document preprocessing steps, etc.*

**4. Software [at least ½ a page]  
*[Note: this is intended to be a high-level description of your software, not the code itself. Separate this section into subsections of (a) code or scripts you have written, (b) code or scripts written by others that you used in your project (with attribution/references). Tables could also be used to summarize the code]****Provide a list of the major pieces of project software and their functionality (general input/output characteristics), both for (a) code you wrote, and (b) code from other people that you used. Feel free to put this information in a table if it helps to organize the information this way.*

**5. Experiments and Evaluation [at least 1 page, preferably 2 or 3]  
*[This is a critical part of your final report and the section where I expect to see the most new material compared to your progress report]****Describe in detail both (a) how you set up your experiments, including what metrics and methods you used for evaluation (test sets, cross-validation, user studies, etc), and (b) what results you obtained (ideally in the form of tables, graphs, etc), e.g., comparing the accuracies different methods and baselines. In this section you can add to your “basic results” by reporting additional comparative results on sensitivity of your approach to different algorithmic choices, e.g., how did performance depend on vocabulary size? On document length? On whether you removed stop-words or not? Did including parts of speech help? And so on.*

**6. Discussion and Conclusion [at least ½ a page]***Discuss what insights you gained from the project. What did you learn about the algorithms you worked with? what results agreed with your expectations? What did not agree with your expectations, i.e., was surprising? What are the major limitations of current approaches to the problem you are trying to address? If you were in charge of a research lab, what ideas and directions might you invest in over the next year or two to try to make major progress on this problem? Feel free to be speculative in discussing possible future directions.*

**7. A separate page on*****Individual Contributions****Each team member needs to submit one paragraph of additional text starting with individual name that provides an honest assessment of which parts of the project you contributed to and which parts were worked on jointly. This should be written individually – you may wish to discuss the plan of what you will write with your project partner, but the paragraph you write should be generated separately by each individual.*