MATH 156: MACHINE LEARNING FINAL PROJECT

LARA KASSAB, PH.D.

This document provides a lot of information. Please read it carefully. Failure to follow instructions will result in lost points. Please contact me with any questions or clarification. If your question is answered in this document, I will refer you back to the document.

1. Summary

The theme of the final project should focus on the application of Convolutional Neural Networks (CNNs) to a specific problem. You should use CNNs. The choice of application, dataset, and specific model is up to you. You will work in small self-selected groups of 2-4 students.

The timeline of the project is:

Group formation	October 11
Project proposal due	October 25
Report due	December 6

2. Project Components

Here are the required parts of the class project along with the corresponding percentage of the project grade.

Group Formation (5 Points). Familiarize yourself with the overall idea of CNNs, why they came about, and their main applications (or some common tasks CNNs are used for). You do not need to know yet all the inner workings of CNNs. We will cover artificial neural networks later in the course, but not convolutional neural networks (this is the goal of the final project). On this google sheet you can write your name, email, and your ML interest to help you find other group members to work on the final project with. You can only access this sheet through your ucla google account. By October 11, you should have your project group decided. Groups should consist of 2-4 students. Submit your group to the assignment on Gradescope. Submit one per group. It is expected that all group members contribute equally, and each member make a technical/mathematical contribution to the project.

Project Proposal (35 Points). I recommend reading this introductory paper on CNNs and discussing it with your teammates. By October 25, you must submit a one-page project proposal. Submit one per group on Gradescope. It should be cohesive and consist of full sentences. It must summarize your proposed project and detail how the work will be divided fairly amongst group members. Remember each group member must make a mathematical/technical contribution. Submission must be in Latex. I highly recommend using Overleaf to collaborate. A latex template is available on Bruinlearn. Here is the rubric:

- 10 Points: Application and dataset are clearly explained.

- 10 Points: Proposal is clear, cohesive, and written in full sentences.
- 10 Points: Proposal outlines a plan for dividing the work fairly.
- 5 Points: Proposal includes relevant references.

Report (45 Points). Your report should be written in the form of a research article where your intended audience is your classmates in other groups. You should write it as if you are talking to other students in the class; you can expect readers know the topics we covered in lecture, but will need a more thorough description of methods/concepts you learned for this project outside of class. Submission must be in Latex. I highly recommend using Overleaf to collaborate. A latex template is available on Bruinlearn. You can have a different outline and section titles for your report, but they should be well-structured and have similar overall components. The final report should be submitted as a single pdf via Gradescope that includes a link to a GitHub repository of the code and all required files.

Project Code (15 Points). The instructor and TA should be able to access the submitted GitHub repository of the code and all the required files to run the code. Your code must be well-documented. You are encouraged to have a readme file. Scientific work must be replicable. Make sure that the results in the report can be replicated by running your code. You must cite your resources (e.g., datasets, codes, models).

Individual Extra Credit (+ 5 Points). If a group member would like to create a YouTube video or a blog post related to the project, they can receive 5 extra points.

3. Report Components

Your report should include the following components:

- Title and author list
- Abstract: This gives a brief synopsis of what is in your report.
- Introduction: This describes, in big-picture terms, what the goal of your project is.
  - Give a background on chosen application. Motivate your problem. Explain why CNNs are a good fit for this problem. Give a brief overview of your proposed approach.
- Background: More precise mathematical details on CNNs.
  - Provide a scientific and mathematical description of CNNs. Cite your reliable resources. Mathematical descriptions, equations, notations are expected.
- Dataset: More details on the dataset and preprocessing.
  - Describe the dataset you are using. Cite the source. Make sure the dataset size and cleaning process is manageable given your time and computational resources (note you can use Google colab for some GPUs). Does the dataset have any major ethical concerns (see lecture slides 1)?
  - Describe the preprocessing steps like cleaning, resizing, normalizing, data augmentation, dealing with missing data, train-val-test split (or cross-validation), etc. Explain and showcase any exploratory analysis of the data that you performed. For example (if applicable), visualizations, dimension reduction, clustering, etc.
- Model: Carefully describe your model.
  - Describe the CNN architecture you used. Did you implement it fully from scratch? If so, describe. Did you modify an existing classic architecture? If

so, which one? Describe it and your modifications. There needs to be a component you worked on or modified.
  - Discuss the training process and all key characteristics including loss function, optimizer, number of epochs, etc. Describe the metrics you used to evaluate the model performance. Remember not to use the test you will report the results on anywhere in your training of your model. Care must be taken in terms of knowledge or data leak (in exploratory data analysis phase, data pre-processing, etc.) or human bias etc.
- Results: Carefully describe the results of your project.
  - Present the performance result of the model on the train and test data. Include any useful visualizations like confusion matrices, accuracy/loss graphs, etc.
  - Write your observations. If a model performed poorly in a certain aspect, describe any idea you have as to why. Discuss potential improvements (hyperparameter tuning, more advanced architectures, better preprocessing).
- Conclusions and Discussion: Summarize in a few sentences the overall goal and results of the work.
- Author contributions: State who did what. Note all group members should make *technical* contributions, not simply writing or putting together the report.
- Acknowledgements: Acknowledge all help and resources from others. This could include the instructor or TA if they provided specific advice, other classmates who gave you ideas or proof read your report, etc.
- References: You must include proper citations with in-text references and full citations in the references section of all sources that you use. This includes (but is not limited to) sources for data sets, methods used, similar work that exists or that you used for inspiration, background work that helps with the big picture, and so on.
- Very important: include a list of the components or requirements described in this document that you could not accomplish or could have done better with more time and one sentence explanation as to why. On the last page of your report, add this list or acknowledge that none of the requirements or components were missing.

General requirements:

- Report length: approx 3-5 pages, maximum of $N+2$ where N is the number of students in your group. Page count does not including references.
- The reader can understand the project and the analysis without requiring outside knowledge on the chosen model, application, etc. You can assume the reader has mathematical knowledge at the level of 156.
- Scientific writing: Everything needs to be specified precisely. All notation must be defined precisely and unambiguously.
- Figures: There should be figures to illustrate your work. This typically includes figures of results, and it often includes figures to help describe the problem (or, when relevant, how a mathematical model or algorithm works).
- You should label the figures and reference them in the text by their number. Captions for figures should be sufficiently descriptive.
- Do not have your figures/tables/algorithms all bunched up towards the end and try to space them in the text appropriately. Tables and algorithms should be well formatted and include descriptive and clear captions.

- Do not include screenshots of the code in your report.
- You must include references.
- You should not copy your project (combination of model, preprocessing, dataset, etc.) from the web (e.g., blogs, kaggle, etc.). You should have original work components.

Here is the rubric:

- 15 points: Report is written clearly and satisfies all the required components described above. See the "Very important" note above.
- 30 points *corrected: Technical and mathematical descriptions, contributions, and correctness of the ML pipeline. Absence of major mathematical flaws and major flaws in handling the data, training the model, and reporting and analyzing of the results.

Formatting can be tricky and takes time. Do not wait until the last minute to put together the report. If this is your first time working with LaTeX, it has a learning curve but you will get used to it. It is a great skill to have so I encourage you all to learn it!