CS598 Projects: Deep Learning for Healthcare

Jimeng Sun

***Abstract*—CS598 Deep Learning for Healthcare is a graduate-level class that teaches practical deep learning methods for healthcare analytics. One major component of this course is a group project, where students will select a paper from the paper pool, replicate the experiments, and report their findings. Ideally, they will also share their codebase and integrate it into an open-source package like pyhealth.**

This document provides the project guideline such as expectations, timelines, and deliverables.

***Index Terms*—Deep learning, Machine learning, Healthcare**

I. Introduction

DEEP learning is now widely used in healthcare applications, thanks to technological advancements like electronic health records, on-body sensors, and genome sequencing, as well as advances in deep learning methods. In this course, you have learned about deep learning methods, healthcare data, and applications of DL methods. Through homework exercises, you have gained knowledge and skills in these areas. Now, you are ready for the next level of challenges - understanding, replicating, and extending recently published works in DL for healthcare. The final results of this project include:

1) **Report** For your project, you will select a paper and replicate the main experiments. You will then write a report evaluating how easy it was to reproduce the results based on a provided checklist. You should also try at least one new experiment that wasn't included in the paper, such as testing the model's sensitivity to hyperparameters or the amount of training data or measuring the variance of the evaluation score due to initial parameters. If you are unable to reproduce the main experiments of a selected paper, you will need to report on your failed attempt. Not all papers are easy to reproduce, and it's okay if your experiments are not exactly the same as the original paper, as long as they are rigorous. In your report, you should identify the questions that need to be answered to reproduce the experiments, or discuss any errors you found in the findings.
2) **Codebase** As part of your submission, you must provide your codebase and code notebook so that others can use your pipeline to reproduce the selected paper. It's best to integrate your codebase into an open-source package like pyhealth, so that it can help other health data science projects and benefit the broader research community.

Both outcomes are acceptable and can earn full credit.

J. Sun is with Computer Science Department of University of Illinois, Urbana-Champaign e-mail: (see https://www.sunlab.org).

II. Teams

Each team can have a maximum of **TWO** students, and individual projects are also permitted. All team members will typically receive the same grade on the project. However, in exceptional circumstances, such as when one team member disappears and does not contribute to the project, grading may be adjusted accordingly.

III. Paper Selection

You should first select **THREE** papers and then narrow down to **ONE** paper (for proposal, draft and final submission) from the provided paper pool [1]. The reason for this coarse-to-fine paper selection is to avoid issues such as no data access, and to increase the success rate of you reproducing at least one of the three papers.

The objective is to assess if the experiments are reproducible, and to determine if the conclusions of the paper are supported by your findings. There are some considerations in choosing the paper to reproduce:

- You should find the problem tackled in the paper interesting.
- You should be able to access the **data** you will need to reproduce the paper's experiments.
- You should choose paper whose **computational requirements** for reproducing the experiment is affordable to you.
- You should not choose a paper we already implemented in the homework.
- While the codebase used in the paper is often available, you should refrain from using it directly. Instead, you are welcome to reference it for guidance, but you should develop your own code for the project.

IV. Timeline

Next, we summarize the timeline.

1) Team formation (1-2 students) & paper selection (3 papers) at this link[2], starting from now.
2) Choose your final paper and fill in your choice here[3] (open the edit option on Mar. 5).
3) Project proposal (PDF), due on Mar. 26
4) Project draft (PDF), due on Apr. 16
5) Final submission (PDF + Presentation + Code), due on May 8

All due at 23:59PM Central Time on the due date. **We do not allow late submission.**

[1] paper pool: https://docs.google.com/spreadsheets/d/19TMVVhRdm2uGNiX1Q9wzsloLatIEcs_iupIQ3VwUHfw/edit?usp=sharing

[2] registration form https://docs.google.com/spreadsheets/d/12TXCnaMKAcUcNA2cmWnaAmzo5j0DD7Nj4qFfsq66JUg/edit?usp=sharing

[3] the editable paper pool (on Mar. 5) https://docs.google.com/spreadsheets/d/19TMVVhRdm2uGNiX1Q9wzsloLatIEcs_iupIQ3VwUHfw/edit?usp=sharing

V. DELIVERABLES

A. Project Proposal (Up to 4 pages write-up + Unlimited references)

Write a project proposal for the selected paper. Your proposal should address the following questions to demonstrate your thorough understanding of the paper you plan to reproduce and effectively communicate its significance to someone who may not have read it:

1) Cite the original paper.
2) State the general problem the paper aims to solve. Do not use the same language as the paper.
3) Describe the new and specific approach taken by the paper. Discuss why it is interesting or innovative.
4) Identify the specific hypotheses you plan to verify in your reproduction study.
5) Outline any additional ablations you plan to do and explain why they are interesting.
6) Explain how you have access to the necessary data.
7) Discuss the computational feasibility of your proposed work.
8) Specify if you will be re-using existing code and provide a link to it, or if you will implement the code yourself.

The proposal should be a single PDF file. We recommend using the project draft template[4].

B. Project Draft (2-4 pages write-up + Unlimited references)

At this stage, you should have fully understand this final-selected paper and implemented all experimental setups and hopefully got some experimental results. You should also have developed your basic code and finished at least one successful run. For the draft, fill out sections from this template [5], replacing the instructions with actual content. For the draft, only complete the following sections, and write "TODO" for all other sections:

- Introduction
- Scope of reproducibility
- Methodology
  - Model Descriptions
  - Data Descriptions
  - Implementation
  - Computational Requirements
- Results - for the draft, results can be any valuable results. For example, results from a simple baseline model in the paper, from intermediate steps prior to the ultimate target task, or from a tiny subset of the dataset. All those followed by your own analysis can be used. Even if your current results are not as good as the ones in the paper, there must be analyses about what possible reasons and solutions/plans are

The draft should be a single PDF using the template provided above. You may also consider using the final project for this Kaggle reproducibility challenge[6] (DDL is Jun 15, 2023).

[4]modify for project proposal https://www.overleaf.com/read/mxprjjmxpmzt
[5]Link to template: https://www.overleaf.com/read/mxprjjmxpmzt
[6]https://www.kaggle.com/ml-reproducibility-challenge-2022-rules

[Notebook Bonus] Along with the project PDF, we also encourage students to add an easy-to-illustrate jupyter notebook (similar style to our homework), including the following contents:

- A summary of the report and findings (Reproducibility Summary, about 200 words)
- An overview of the data with any helpful charts and visualizations from the report and ideally directly using the dataset in the notebook (maybe link to the data folder or URL).
- An overview of the methodology and experiments run, ideally with executable code examples
- A summary of the key results.
- A references section.

[PyHealth Bonus] We encourage students to contribute to the PyHealth. From the project, if you feel the dataset, the task, or the ML model can be a great addition to the pyhealth package, we encourage you to clean up the code:

- **new dataset:** please follow the existing dataset class structure[7]
- **new task:** please following the existing task function structure[8]
- **new ML model:** please following the existing model class structure[9]

and send a PR to pyhealth "develop" branch.

We will process the PRs one by one following the submission order (we prefer new datasets and models), so please take action now (do not wait until the end of the semester). If we feel your contributions are valuable, we may work with you to integrate the PR into pyhealth. **Remember, the bonus score will be granted only if the PR is merged.** You can understand our code structures from live videos, colab tutorials, and function descriptions, all on the website[10].

C. Final Submission

The final submission includes the final report, presentation, and code for your final-selected paper.

1) Final Report (4-6 pages write-up + Unlimited references): This should follow the same format as the draft, but all sections should be filled in.

2) Presentation (Up to 4 minutes): You should prepare PPT slides that clearly illustrate the main points in your work and the main results. Good visuals are important here – the presentation should be eye-catching, clear, and self-contained. Assume your audience has the background given in this class but remember to spend considerable time introducing the motivation and setup of the problem you are addressing. You should also spend time comparing your reproduction attempts with what the paper showed and (if you could reproduce) the additional ablations.

[7]https://github.com/sunlabuiuc/PyHealth/tree/develop/pyhealth/datasets
[8]https://github.com/sunlabuiuc/PyHealth/tree/develop/pyhealth/tasks
[9]https://github.com/sunlabuiuc/PyHealth/tree/develop/pyhealth/models
[10]https://pyhealth.readthedocs.io/en/latest/

Please upload a *unlisted*[11] video on YouTube of your presentation (demo by one representative or multiple students, set an access key if you want) to share with us. Put the YouTube link under the report title.

3) Code: Publish your code in a public repository (e.g. on GitHub, GitLab, BitBucket). Make sure your code are documented properly. A README.md file describing the exact steps to run your code is required. You can refer to the ML Code Completeness Checklist to write the README file and make sure your code submission is complete. See this blog post on best practices for reproducibility.

VI. Grading Scheme

The entire project has 100 points and will account for 40% of your class grade. The specific points are as follows:

- Proposal (10 Points)
- Draft (20 Points)
- Final Report (50 Points)
- Presentation (10 Points)
- Code (10 Points)
- Notebook Bonus (additional 5 points)
- PyHealth Bonus (additional 5 points)

Good luck to everyone!

Acknowledgment

Part of the project instruction is adapted from CS662 Advanced NLP at the University of Southern California.

[11] See https://support.google.com/youtube/answer/157177