**CS 441 - Final Project Assignment**
**Spring 2023**
**Due May 3 11:59pm**

You have three different options for a final project:

1. Perform a a pre-selected **challenge**
    - Team Size: 1-3
    - Grade is on completion, performance, and quality of proposed solution and experiments
2. Select your own **benchmark** task
    - Team Size: 2-4 (4 requires comparing approach on at least two benchmarks)
    - Grade is on motivation/background, completion of approach, performance, quality of proposed solution and experiments
3. Create your own **custom** task
    - Team Size: 2-4
    - Grade is on motivation, completion, and quality of data preparation, approach, and experiments

**Challenges**

1. **Detect birds in drone data**: https://www.mva-org.jp/mva2023/index.php?id=challenge
   Note that the submission leaderboard closes on April 14, so you can use the train 2 set or a validation set for evaluation. Vision tasks typically require a lot of storage and computation, so this is not recommended, unless you have access to good GPU machines.

2. **Disaster tweets**: https://www.kaggle.com/competitions/nlp-getting-started

3. **House price regression**:
   https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques

**Benchmark task**

This means you are using a published dataset, but not one of the above challenges. Part of your project is summarizing two or three state-of-art approaches on the benchmark. A good source for finding benchmarks of interest is https://paperswithcode.com/datasets. Your goal is to implement a good solution, which can be based on published methods, or you can try out a new idea.

**Custom task**

This means you are creating your own task, and not using a published dataset or challenge. This could be based on a personal project, or a project with your research group. Your goal is to get something working, though there may be room to improve the method.

## General Guidelines

The intent of this project is for you to plan, implement, and analyze an ML task from start to finish, making use of available tools and the techniques you've learned. You can use pretrained models and/or machine learning libraries. You cannot directly use a solution that is customized for your challenge or benchmark. For example, you can use HuggingFace language models for the Disaster tweets challenge, but should not use a repo developed specifically for that challenge.

## Deliverable: Report + Code zip

**Layout/Format**: Up to 3 pages, size 11 Times New Roman font. Latex, Google Doc, or Word is ok. Latex is easiest for table/equation formatting if you have experience with it. Custom task can use up to 1 additional page if needed to explain/visualize task. Convert report to pdf before submitting.

**Content**

- Motivation: *Required for benchmark/custom (not challenge).* Describe the problem of your benchmark or custom task and why it's important. [¼ page, maybe more for custom task]
- Background: *Required only for benchmark*. Briefly summarize 2-3 state-of-art approaches to the benchmark. [¼-½ page]
- Approach: *Required for all projects.*Describe your final approach, ideally with enough detail for an ML expert to re-implement, but taking no more than 1 page. For custom task, also describe how you collected data, if applicable. [½-1 page]
- Implementation Details: *Required for all projects*. List packages that you use (excluding basic ones like numpy) and external sources of code. Describe parameters/architecture.
- Experiments: *Required for all projects.* Include results of your approach and ablations indicating importance of design parameters. For benchmark task, report results of methods discussed in background. [½-1 page]
- Discussion: *Required for all projects*. What are your conclusions? What would you do differently next time? [¼ page]
- Citations/Acknowledgements: *Required for all projects*. Cite sources of ideas and code, including websites/blogs/github pages and models. You do not need to cite standard libraries.

**Rubric**

- Report [up to 20]
    - Well written and organized, clear, complete [20 / 20]
    - Some parts missing or unclear [10 / 20]
    - Very incomplete or completely missing [0 / 20]
- Approach [up to 40]

    - Well-justified, appears to be correctly implemented, some innovation [40 / 40]
    - Good approach but straightforward adoption of current techniques [35 / 40]
    - OK, but a better approach could be found with some effort [30 / 40]
    - Significant flaws in approach [0-20 / 40]
- Experiments [up to 40]
    - Clearly presented, with good discussion: +15
    - Good investigation of design through ablations / comparison: +15
    - Excellent performance: +10

Also submit a link to your code (or code directly if not too big). Teams must also submit a statement of contribution in the text. Each team member should submit the same report and code, but may individually report the team contributions.

For the **statement of contributions** (required only for 2+ person teams), in the text state: "All team members contributed equally" or "Team members did not contribute equally" and list the contributions of each team member. E.g

All team members contributed equally.
* Billy Joel: Researched approach, contributed equally to coding, took lead on motivation and approach sections of report.
* Elton John: Contributed equally to coding, recorded experimental results, took lead on implementation and experiment sections.