[CMPUT 466/566, Winter 2023] Machine learning

Course Project Description

Instructor: Lili Mou
LMOU@ualberta.ca

Objectives:

1. [10 marks] The basic goal of the mini-project is for the student to gain first-hand experience in formulating a task as a machine learning problem and have a rigorous practice of applying machine learning algorithms.
2. [5 marks] The second goal (optional to undergrads) is to accomplish a non-trivial machine learning project, such as replicating a recent top-tier machine learning publication (published at ICML, NeurIPS, ICLR, etc.), proposing new models, and empirically analyzing machine learning models in a significant way. Replicating a paper published at an unknown or non-machine learning venue may not constitute a non-trivial project. The 5 marks count as bonus for undergrads but are included within 100 total marks for grads.

Note that only one project is expected. A non-trivial project must also satisfy the basic requirements.

Team work

Collaboration of the course project is possible only if

1) all team members have already had first-hand experience,
2) they intend to do a non-trivial project, and
3) the team must have no more than three members.

Online lab sessions are a good opportunity for students to know each other and form collaboration teams. The team has to apply in NOI before the NOI deadline. The application may be declined if any of the team members does not have adequate machine learning background.

If teamwork is approved, the team members (name, ID, and email) and individual contributions must be stated clearly in all submissions. All team members must upload the submissions to their own eClass assignments.

Timeline and submissions

All due time in this section is in Edmonton time. There will be another 24h grace period.

A project intended to satisfy the basic requirements only (10 marks) does not need to submit the notice of intent or a proposal. They only need to submit the final project, and the deadline is ~~11:00AM Apr 18 (extended to 11:00AM Apr 20)~~ Apr 20 (extended to Apr 25).

A non-trivial project requires significantly more time than a project satisfying basic requirements only, so a significant amount of time has to be set for the project. It must follow the mandatory timeline:

- Jan 19 (extended to Jan 24): Notice of intent
- Mar 2 (extended to Mar 7): Proposal
- Apr 20 (extended to Apr 25): Final report

All deadlines are by 11:00AM (Edmonton time).

The student must decide early if he/she is going to do a non-trivial project. If so, the student must send a notice of intent (NOI) by the deadline, indicating a title and a short description. The NOI will not be reviewed but is mandatory for a non-trivial project. If several students intend to form a group, the NOI must also include every team member's name, email (ccid), and his/her prior experience in machine learning. The approval of teamwork will be based on students background and the intended topic.

The student is supposed to read literature and prepare experimental environments after NOI. By the proposal deadline, the student must submit a pdf proposal to eClass. The instructor will read the proposal and make a comment, especially on how non-trivial the proposal is.

Notice that an intended non-trivial project may not get all 15 marks or, if not satisfying the basic requirements, may not even get 10 marks.

Basic Requirements [10 marks]:

- *Formulating a task into a machine learning problem*. The student CANNOT re-use any task in coding assignments (namely, house price and MNIST datasets) as the course project.
- *Implementing a training-validation-test infrastructure, with a systematic way of hyperparameter tuning*. The meaning of "training," "validation," "test," and "hyperparameter" will be clear very soon.
- *Comparing at least three machine learning algorithms.* In addition, include trivial baselines (if possible). For example, a majority guess for *k*-category classification yields *1/k* accuracy. The machine learning algorithms must be reasonable for solving the task, and differ in some way (e.g., having different hyperparameters does not count as different machine learning algorithms).

General machine learning packages may be used for the course project. However, the student cannot use the codebase specific to the task at hand and run a few scripts like "sh run.sh".

Requirements for a non-trivial project [5 marks]:

A non-trivial project could be either replicating a recent, sophisticated machine learning paper, proposing new models, or conducting empirical analysis of machine learning models in a significant way.

Typically, a non-trivial project involves a significant amount of literature reading, programming, and experimentation. A student would not expect any additional marks by trying some CNN/RNN models, or applying existing code base to a new task in a straightforward way. If a student seeks non-triviality marks by replicating a recent paper, the student should assume the code base of that paper does not exist.

If a student has doubts about how non-trivial the project is, the student may check how much mathematical and algorithmic formulation there is.

Final report submission:

The submission must contain a PDF report **and** the code to reproduce the results. (Non-complying file format will result in mark deduction.)

The code can be submitted by either a zip or an online, public repo (without logging in or accepting invitation). Notice that a private GitHub repo with an invitation is not accepted.

The format of the report is flexible, but generally, the report should contain

- A short introduction, describing the background of the task
- Problem formulation (what is input, what is output, where did you get the dataset, number of samples, etc.)
- Approaches and baselines (what are the hyperparameters of each approach/baseline, how do you tune them)?
- Evaluation metric (what is the measure of success, is it the real goal of the task, or an approximation? If it's an approximation, why is it a reasonable approximation?)
- Results. (What is the result of the approaches? How is it compared with baselines? How do you interpret the results?)

Grading criteria:

Basic requirements [10 marks]:

- If the submission is not a machine learning problem, then 0 marks.
- Otherwise, the grading starts from 10 points. If one or more of the above requirements are not fulfilled, it will result in mark deduction for one or a few points.
- Presentation enters the mark in a multiplicative way. The factor is 1 be default, if the report is reasonably well written.

Non-triviality [5 marks]: Marks will be distributed to literature review, proposed approach, and experimentation.

Tips:

1. The course project only counts 10--15% of the total marks, and obviously, this course focuses more on math derivations than coding. It is more important to formulate a machine learning system in a rigorous way and complete the project in time than do a super fancy project (which may require too much work and has a risk of not being finished in the course timeline).
2. In fact, many students sought minimal efforts to obtain non-triviality marks in the past, which is not possible. In general, not many students got the 5 marks and students should not worry about it. Even for graduate students, getting a 5-mark deduction is not a problem, because the letter grade cutoff will be adjusted accordingly and may be different from undergraduates.
3. Using external general-purpose machine learning packages is allowed but should be acknowledged (e.g., use libsvm to solve the task by a few lines of function call). However, using a code base directly related to your task is not allowed (e.g., download a GitHub repo and only write a few lines of script like "sh run.sh").
4. There is no constraint on the number of pages of the course report. However, the length should reflect the substance of the project, and in a normal case, a few pages suffice. An over-lengthed report will not yield a higher mark. On the contrary, it shows poor presentation skills (and may lead to mark deduction). The report must be written in text with results organized in an appropriate form (such as

tables and figures). Python notebook, code snippets, and program output are **not** considered as textual report.

5. We will grade the course project in a lenient way. However, we do not accept mark negotiation. The basic requirements are clearly stated above. The instructor will adjudicate the degree of non-triviality based on the same criteria applied to all students.

Collaboration and future opportunities with instructor

The instructor's research interest is mainly in natural language processing, especially focusing on unsupervised text generation and latent structure reasoning for text understanding. Students are welcome to pick one of the topics as their non-trivial course projects and are encouraged to read the related papers. If the student is indeed interested in the project, a discussion with the instructor is mandatory in this case.

The instructor is looking for MSc students as well as undergrad RAs. If a student is interested in being supervised by the instructor for future research, a course project is an ideal starting point. The instructor is also willing to offer Individual Study courses in future semesters.