EECS 545 FA 2021 Project Guidelines

We recommend reading this entire document before starting to think about your project.

Details will continue to be added to this document.

Any questions? Please ask in Piazza.

Acceptable Projects

The project can take one of the following forms:

- **[Implementation Track] Implementing a machine learning method from published papers.** You can select a **recent** (last ten years) journal or conference paper that was published in JMLR, PAMI, ICML, NeurIPS, AISTATS, ICLR, AAAI, or IJCAI, on a recent ML method/algorithm, work on understanding how the method works, and implement the method on real data (not provided by the paper) to illustrate your understanding. You can additionally verify your implementation is correct by comparing with the reported results on the dataset used in the paper. The report will need to explain the method, describe how it is related to methods covered in class, and quantitatively and qualitatively assess its accuracy and performance. For a paper published in a different venue than listed above, inquire on Piazza. Unpublished papers are not acceptable.

- **[Application Track] Applying machine learning concepts learned in the course to a novel dataset.** A dataset is considered novel if it has not been extensively analyzed using modern machine learning techniques. This allows for datasets that may have been published in an application oriented journal, or that have only been studied using one or two basic ML methods. *The project must apply methods that have not been previously used to analyze the given dataset.* This track is envisioned for students, such as PhD students, who are very interested in a particular application domain, and have access to data that is novel or has received only a little attention in the ML literature. The dataset should be unrestricted, non-proprietary, and not subject to IRB (Internal Review Board). The analysis will have to include a comparison of **at least 3** different machine learning methods. Selecting multiple tuning parameters for a single method does not correspond to different models -- the methods you select need to be qualitatively different The report should include a description of any exploratory data analysis or data cleaning, should employ model selection where appropriate, and display visualizations of the data to aid the reader in understanding its unique features and the challenges presented.

- **[Open-Ended Track] Developing a new ML algorithm, or making a substantial improvement on existing machine learning algorithms.** This can be open-ended and thus challenging; this should be at a research level and potentially publishable. You should try to add novel contributions in terms of methodology, technique, analysis, etc, and deliver some new findings. A natural approach is to combine two existing learning algorithms to come up with a non-trivial, better algorithm.

IMPORTANT NOTES:

- No student may do a project on the same topic as their literature review.
- Every group member should make a roughly equal contribution to the project. In addition, each group member should have an active role in understanding and implementing the core methodology. This requirement is to avoid situations where one group member spends all of their time doing menial tasks.
- A few groups every year choose to investigate deep neural networks, which is fine. However, based on past years, deep learning projects are the projects that students often have the most difficulty completing. This is because of the intense data and computational resources needed, as well as the need for expertise in the fine art of learning neural networks. Even if you code everything perfectly, it may take days or weeks to run your code on available machines, which is obviously problematic. We just want you to be aware of this issue, and to choose your project carefully if you elect to study deep learning. Avoid recent papers produced by corporate research groups (e.g., Google, MSR) as they undoubtedly have far more computing resources than you will.
- We expect that ARC-TS will provide 545 students with free accounts on the Great Lakes Computing cluster, together with a computing budget (details forthcoming). The cluster supports regular CPUs and also GPUs. Note that GPU programming can be tedious so factor in the time needed to learn the system, and plan to get started early. Students may also consider Google Colab which offers users access to one free (albeit slow) GPU.

Deliverables and Grading

- Project proposal (10%)
- Final report (80%)
- Peer grading (10%): Both the proposal and final report will be assessed using peer grading. Each proposal/report will be assigned several reviewers, and the median peer score will be the assigned grade. This is a required aspect of the project and points will be deducted for non-participation, or for low-quality or unfair reviews. Projects whose scores have a high variation will be automatically inspected by course staff. In addition, students will have the opportunity to call our attention to sub-standard reviews. All group members will receive the same grade unless it is evident that different members contributed substantially different amounts.

Project Proposal Guidelines

The proposal document must be in a PDF file, generated by LaTeX (no MS Words, etc.). It must be written in NeurIPS format. Please follow the formatting instructions for NeurIPS 2021. The proposal document **must be anonymized**: use "\usepackage[final]{neurips_2021}" . The proposal should be **2-4 pages**, including figures/tables, and excluding references.

Your proposal should include the following sections (the order can be changed):

1. **Title**: A descriptive title. Make sure the document is anonymous. Just below the title, indicate the track and also the size of your group. Use the \author command for this. Be sure not to reveal author identities. For example, \author{Application Track, 4 Team Members}.
2. **Introduction or Problem statement:**
    a. What is the topic or goal of your project? Why do you think it is interesting or worth pursuing? Why do you think it is not trivial?
    b. What would be the novelty or contribution of your project?
3. **Method**: Proposed approach or methods to be developed
    a. [For the Implementation track] Explain the method (the high-level idea and optionally the technical part) in your own words, and background as much as needed (assuming the reader would have understood the topics covered in this class). A blind copy-and-paste is not allowed. Discuss the attributes of the method that warrant its use.
    b. [For the Application track] Which machine learning methods are going to be investigated?
    c. [For the Open-Ended track] Explain the method or approach you are developing. It is fine to not have a complete description.
4. **Related work**:
    a. Discuss related work. A comprehensive survey of related works is not necessary for the proposal. Elaborate on how cited references are different from or related to your project.
    b. For example, projects in the Application track might discuss relevant works on the dataset, similar analysis. Projects in the Open-ended track should discuss relevant prior approaches and discuss how they are different from and related with your method.
5. **(Plan of) Experiments**
    a. How will you **implement** your work? List some software to be used or developed. What software or off-the-shelf tools can be used? What are you going to create on your own?
    b. **Dataset** to be used (real-world data or simulated). You can discuss the nature of the dataset. How can it be obtained publicly, or how do you collect/generate the

data? What are the statistics of the dataset? Why is it challenging, or why would it be useful/interesting to study?

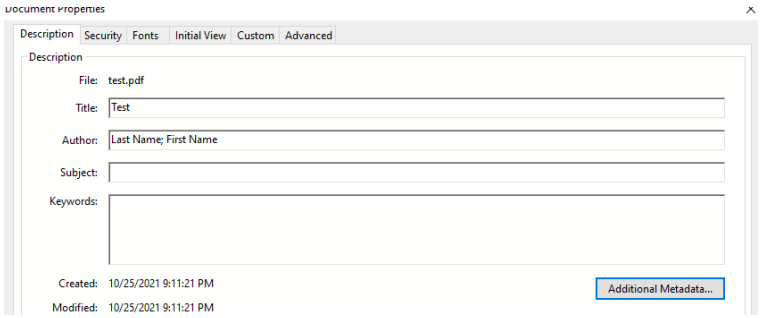
c. **Evaluation**. How will you evaluate the methods considered? What experimental settings, datasets, and evaluation metrics will you use?

6. **Plan of Project**
    a. Anticipated division of work over team. Please avoid using real names, and write Author 1, Author 2,... instead.
    b. Explain any experience or expertise you will be drawing on to complete the project successfully.
    c. What challenges and difficulties would you expect?
    d. Include milestones, to increase the likelihood that you are able to deliver at least some positive results.
    e. What are your contingency plans in case something goes wrong?

Proposal Submission Instructions on HotCRP

Things to check before the submission:

- Make sure you are using the NeurIPS template, and the proposal should be ***anonymized***. To do so, in the .tex file, use the following commands.
    - \usepackage[final]{neurips_2021}
    - \author{Your Project Track, Number of team members}, e.g. \author{Application Track, 4 Team Members}.
- Remove author information in the metadata of the PDF file. Note that the metadata of a PDF file may reveal authors' information. For example



  To remove metadata from a PDF file, you could use Acrobat or online tools.

- ***2-4 pages*** including figures/tables and excluding references.

You only need to create **one** submission per team, i.e., only one member in your team needs to submit the proposal. Open the submission site https://umich-eecs545fa21.hotcrp.com/, sign in

with your umich email account and click the **New submission** button. Then, enter the following information about your submission, including:

- **Title** --- Title of your project
- **Submission** --- Upload the proposal .pdf file
- **Authors** --- list all the team members.
    - In the “Email” field, fill in the umich email accounts. “Name” and “Affiliation” files will be auto-completed afterwards.
- **Contacts** --- Optional
- **Track** --- Which category does your project belong to?
- **Honor Code**
- **Checklist** --- Double-check your submitted .pdf file and check these entries.
- **PC conflicts** --- Conflict of interests could arise in any situation where a person’s ability to evaluate fairly and without bias is compromised. For example, this could include your close friends, teammates, lab mates, family members, etc. Select from the list all that apply to **any** members in your team.
- Email authors --- Optional.
- **IMPORTANT:** please check the checkbox **“The submission is ready for review”** which will appear only after you click on "Save Draft." If this box remains unchecked, your submission will only appear as a draft. So, click on **Save Draft,** then check **"The submission is ready for review."** in order to submit. Afterwards, you can still update your submission until the deadline.

You are all set! Please double check whether everything looks good. Note that you may receive warnings like “PC conflicts: *You may have missed conflicts of interest with xxx*”. Please ignore this warning as this is simply caused by the same affiliation.

Proposal Peer Review Process/Rubric

Each student will review 3 proposals (and 3 associated final reports) that are randomly assigned among non-conflicting reviewers. The review process is double-blind: neither the reviewers nor the authors will be aware of each other's identities. You will assess the same 3 projects for both the proposal and the final report. This allows the authors to address any concerns raised during the proposal. Review assignments will be released approximately one day after the submission deadline. Please sign in to the HotCRP site later on to check out the projects that have been assigned to you.

The first part of the review is a single text box where you need to summarize the project document you reviewed. Then, the review form will ask reviewers to assess how closely they agree with each of the following statements, on a scale of 1 = strongly disagree, 2=disagree, 3=neutral, 4=agree, 5=strongly agree.

- Formatting: The proposal adheres to the formatting guidelines

- Writing quality: The document is well organized and clearly written. It is easy to follow and understand. Math notations and phrasing are accurate and technical. No obvious errors or typos.
- Track adherence: The paper adheres to the expectations of the track (see first page of final project guidelines)
- The problem under consideration is clearly stated and motivated
- At least three key references are provided and their relevance to the project is explained.
- The methodologies and data analysis techniques are outlined at a high level
- The evaluation process, including data sets, existing code repositories, and performance measures, are clearly described
- The project plan is clearly stated, and describes how the work will be divided, milestones, and contingency plans if technical difficulties arise.
- The amount of work seems appropriate for the given group size

In addition, the review form will include a single text box where reviewers should write some summary statements reflecting the strengths and weaknesses of the proposal, and justifying any low scores. Constructive criticism is of course welcome.

Reviewers please remember: Low quality reviews may lead to a penalty in the peer review component of your project grade. It is important to provide constructive, thoughtful reviews.

Final Report Guidelines

Formatting Requirements:

- 8 page limit using the NeurIPS 2020 template, including all text, figures, and tables.
- Font size and margins: these are determined by the NeurIPS template
- Give your project a descriptive title, not something bland like “EECS 545 Final Project”
- Just below the title, indicate the track and also the size of your group. Use the \author command for this. Be sure not to reveal author identities. For example, \author{Application Track, 4 Team Members}.
- Include an abstract: A brief overview of the report
- Unlimited bibliography. The bibliography should be formatted in a consistent style.
- Unlimited appendices, which should come after the bibliography. This is optional. Authors may reference appendices from the main report, but reviewers are not expected to read appendices.

Content: Every report should address the following (but don't feel obligated to make these the section headings or to cover the topics in this order):

- Motivation. This will depend somewhat on your track. For example, for the implementation track, describe the motivation behind the paper whose method you are implementing. For the application track, a lengthier description may be appropriate:

describe the application under consideration, and why existing machine learning approaches, if any, are inadequate.

- Problem statement, both conceptual and mathematical. For the application track, if the problem is one already covered in class, this may be brief.
- Related work. Review the relevant literature needed to properly motivate and explain the significance of the problem your work addresses.
- Methodologies implemented/applied/developed, and the rationale behind them. Include supporting implementation details or theory as you deem fit and as space allows. You should try to teach the reader as if you want them to be able to understand and implement the method for themselves. Mention any necessary data cleaning or pre/post-processing. For the application track, at least 3 ML methods must be applied, in addition to data cleaning, model selection, or visualization that may be required.
- Evaluation: You should conduct experiments on real (and possibly synthetic) data. If you use synthetic data, e.g. for a reinforcement learning project, the data must be generated by a reliable, independent, third party - e.g. an open source multiphysics simulation package or an open source gaming software package. For the implementation track, it should be a data set not covered in the original paper. For the application track, it should be a data set that has received little attention in the ML literature, and the methods you apply must have not been applied to this data before. Discuss performance measures used, results, parameter tuning, design choices, etc. We hope to see quantitative, objective evaluations. Comparisons with natural competitors is a plus. Ideally, someone should be able to reproduce your work based on what you describe.
- Conclusions: What did you learn? What were the methods' strengths and weaknesses? What were the project's success and failures? You need not report only positive findings, you may conclude that a method was not as successful as advertised or that your data application results were not as good as anticipated due to missing data or other complications. Negative resul;ts also push the science forward as long as you can explain them and extract a lesson learned from them.
- Bibliography: A list of references cited in the report.

Advice

- Define all terms and variables. Keep in mind that even if a term is a very simple concept in a certain field or application domain, do not assume others are familiar with it.
- It is not necessary to show that your chosen method is superior to others. Your conclusion may indeed be that a method has some weaknesses that were not highlighted in the original publication.
- Your audience is your classmates, so you may assume anything that we have covered in class.
- Every group member should contribute equally, and every group member should contribute in a way that involves learning about new machine learning techniques. If you

think another member of your group has not put forth a reasonable effort, you may let us know by a private Piazza note and we will take that into consideration.
- Even if different group members work more or less independently on different parts of the project, the final report should read as a unified whole. How do the different parts relate to and inform the others? All group members should participate in writing the final report.
- We encourage the use of figures to make concepts visual and concrete. Figure captions should state what the reader should conclude from the figure and they should be pretty much self contained.

Submission:

- Upload your final project report to [LINK FORTHCOMING]. Please create a new submission. It will be assigned to the same reviewers who reviewed your proposal. The review form will be different, however, as indicated above.
- In addition to the project report, every group should submit a single zip file containing all source files of computer programs written by group members. While these files will not be graded, and we won't try to run your code, we ask you to submit this work in case some question arises during grading that the code would help to clarify. The code should be uploaded as an attachment to hotCRP.com when the final report is submitted.

Final Report Peer Review Process/Rubric

The review form will ask reviewers to assess how closely they agree with each of the following statements, on a scale of 1 = strongly disagree, 2=disagree, 3=neutral, 4=agree, 5=strongly agree.

- Formatting: The paper adheres to the formatting guidelines
- Track adherence: The paper adheres to the expectations of the track (see first page of final project guidelines)
- The problem under consideration is clearly stated and motivated
- The literature review is appropriate
- The methodologies and data analysis techniques are clearly explained
- The evaluation includes appropriate detail about the experimental setup and quantitative performance measures where appropriate
- The conclusion summarizes what was learned and describes the pros and cons of the method(s) studied.
- The amount of work seems appropriate for the given group size

In addition, the review form will include a single text box where reviewers should write some summary statements reflecting the strengths and weaknesses of the report, and justifying any low scores.

FAQ

- How can I find teammates?
    - Attend class and talk to your classmates, or post under “Find Teammates” on Piazza.

- I’m auditing the course. Can I participate in the group project?
    - No

- Where can I find some publicly available datasets?
    - UCI Machine Learning Repository
    - physionet
    - awesomedata/awesome-public-datasets: A topic-centric list of HQ open datasets.
    - List of datasets for machine-learning research
    - https://www.kaggle.com/datasets
    - https://crawdad.org/

- Can we use external libraries (e.g. scikit-learn, TensorFlow, PyTorch, etc.?)
    - Yes, but it must not make your project trivial. For example, it would not make sense to just use someone else’s code for an Implementation-Track project. For a deep learning project, you can use PyTorch and similar packages, but you should define all layers yourself. For non-deep learning projects, it may very well make sense to implement everything from scratch. Keep in mind that the effort level needed to complete your project will be a factor in its scoring. If it does not take that much effort to implement and/or apply the method(s) you have chosen, then you will need to find a way to add more work. For example, you could study two papers, or implement a competing algorithm for comparison, or develop some extension.

- Can I do a project by myself?
    - No. Please work in groups of 2-4 students.

- Can my group use R or Matlab instead of Python?
    - Yes, if you feel it will lead to a substantially better project.

- What qualifies as “recent” paper for the Implementation Track?
    - Anything in the past 10 years is fine. Earlier than that, just ask.

- What are some ideas for topics?
    - Active learning (when labels cost something and the label budget is limited)
    - Semi-supervised learning (how to use unlabeled data to help design a classifier)
    - Novelty/anomaly detection (classification when one class has no training data)
    - Multi-label learning (when patterns have multiple valid labels)
    - Multi-view learning (learning with two disparates sources of information, e.g. audio and video, or speech and text)
    - Nonparametric Bayesian methods (e.g. latent Dirichlet allocation for topic modeling)
    - How to handle missing data (when some features are not observed)
    - Robust methods (for PCA, SVMs, etc.)
    - Topics in reinforcement learning (adaptively controlling an agent in an unknown environment)
    - Advanced methods for nonlinear dimensionality reduction
    - Random projections and randomized algorithms for learning
    - Multi-task learning (learning several related classification tasks at the same time)
    - Transfer learning (when training and testing distributions differ)
    - Online learning (data arrive sequentially)
    - Ranking and recommender systems
- My group and I are doing a project in 'Implementation Track'. The paper we are referring to has a github repository and the code is available. We are using a completely new dataset (on which the project has not yet been tested). Our implementation is based on the original paper (Github code), and we are making the necessary changes for the new dataset as well as modifying few functions according to our implementation. This being an implementation track project, is it obligatory to write the entire implementation of the paper from scratch or can we refer to the github repo?
    - The answer depends on how much work it takes. The idea of the implementation track is generally to implement a method yourselves. If you are finding that to do a complete re-implementation would be too much work, you can use pre-existing code for portions of your implementation. In that case your report should clearly explain what parts you coded and what you borrowed, and try to convey how much work that took. Keep in mind the rubric contains an item where the reviewers will assess whether the overall amount of effort required was appropriate. So you will want to write a brief section to convey this clearly. You will also be asked to submit your final code for inspection if necessary.
- The "motivation" subsection we need is pretty much covered by the introduction in our original proposal, is it okay to just leave that in there (effectively) word for word?
    - Yes, you can reuse text from your proposal in the final report. Thanks for asking.
- I was curious as to what the problem statement section of the final project report entails. Should it just be a brief defining of terms/some fundamental mathematical definitions relevant to the problem like in the lit review or should we go more in depth into the math behind our algorithm in this section?

    - Essentially any good ML methodology paper leads off by making three points. Each of these could correspond to a paragraph but not necessarily. (1) Description of a machine learning problem, and why it is important. (2) High level description of existing methods and their deficiencies (this is not your literature review, that is more extensive) (3) Contributions: How the proposed method addresses one or more of the deficiencies in 2. These are all high level introductory descriptions to set the stage and motivate the work. For the application track, rather than focusing on a novel method, the focus would be on a novel data set and what are the challenges of analyzing that data set and/or the potential rewards of doing so. After the introduction, a more precise mathematical formulation of the problem would be appropriate. There is no one right way to structure a report. You can look at different papers to get ideas.
- For the Application track, do distinct neural network architectures count as different methods?
    - Different families of architectures (VGG, Resnet, etc.) count as different methods, but different architectures within the same family do not.