Project Instructions for Neural Networks and Deep Learning (CSC413) (Winter 2024l)

Amir-massoud Farahmand, Amanjit Singh Kainth, Robert (Rupert) Wu
Department of Computer Science, University of Toronto

1 Introduction

The goal of the project is to give you an opportunity to gain experience in doing research in neural networks (NN) and deep learning (DL).[1] The range of acceptable research topics is wide. Your work can be application-oriented, algorithmic, empirical, or a combination of them. This is discussed more in Section 2.

The project contributes to 30% of your final mark. It has two components. The percentage of each component and the deadline for each of them are (subject to minor changes)

- Proposal (10%): February 27th (Section 3)
- Report (and Source Code) (20%): April 19th (Section 4)

Collaboration. The project should be conducted collaboratively as a team of 3 or 4 members. Everyone should contribute to the project in a meaningful way, and they should be clear about their contributions.

You can use a machine in the form of copilot, ChatGPT, etc. to help you in the project, but that would be considered as a group member. We require that each team has at least three (3) human members. For example, you can have a 4-human group or 3-human + 1-machine group, but not 4-human + 1-machine or 2-human with 1-machine. You should treat the generative AI machine that helped in your project, beyond doing the computation required to generate empirical results, the same way as a human in reporting its contributions to the project.

Some frequently asked questions are answered in Section 5.

2 The Many Varieties of Projects

You may choose any topic of your choice, as long as it is related to NN and DL. We describe a few general directions you may want to pursue, and how you can choose a topic within them. Your project might be a combination of these general directions.

As a general guideline, you do not need to invent a new algorithm or achitecture, reach a state of the art performance, or completely solve a new application domain to be successful in this project. We realize that there is not much time in a semester to learn enough about a new research area to innovate a new idea and execute it completely (though sometimes your fresh perspective might lead to an idea that others have not thought before. In that case, you have to cherish that opportunity and pursue it). The goal is to give you a taste of what research in NN and DL might look like. If you feel delighted enough after the end of this course, you have the option to continue working on the project

[1] This document may be slightly changed in the coming weeks in order to clarify any questions that you have. This version: 2024 February 10.

Preprint. Under review.

with your team. This being said, you have to spend a good amount of time on this project and you must follow rigorous scientific methodology in pursuing your research.

When choosing your project, you have to consider whether there is data available for you and you have access to enough compute power for that project. Since the project deadline is only in two months, consider tailoring your project ideas to what data and compute resources are available to you.

Before providing specific suggestions for each type of project, we have a general advice on how you can start if you do not have any idea already: You can consult the proceedings of top AI/ML venues such as NeurIPS, ICML, ICLR, COLT, AAAI, IJCAI, AISTATS, and journals such as JMLR, TMLR, MLJ, and JAIR, to find many interesting papers. A paper may catch your attention. Read it carefully. Afterwards, read some of the papers that are cited within that paper, and backtrack. You often find the same set of papers referred to again and again. It is also very helpful to see what other more recent papers have cited your originally selected paper. This helps you figure out what advances has been make since that original paper.[2]

2.1 Application

If your main research topic is in an application area, broadly defined, you can investigate whether you can formulate it as an ML/DL problem and solve it using NN-based algorithms.

For example, if you work on healthcare, robotics, user modelling, fraud detection, product recommendation, speech recognition, natural sciences, you can probably find a problem that can be formulated as an ML problem. Of course, there are many application domains that can potentially fit well within the ML framework, but have not been explored much yet. Discovering this possibility is an exciting endeavour.

If you decide to go through this path, you should collect appropriate dataset, compare several DL algorithms that we have covered during the course (or some new ones that you find in research papers), and compare them with each other. You need to follow high standards of empirical evaluations.

2.2 Empirical Study

Empirically investigating an already existing algorithm is a reasonable project. Think of yourself as an experimentalist who wants to understand the behaviour of an algorithm through careful design and conduct of experiments.

You want to know when the algorithm works and when it does not. The original paper that introduced the algorithm might have not explored all relevant questions. It is possible that their authors only reported successful results. Your goal is to empirically investigate the range of problems and conditions that result in success and failure of an algorithm.

You need to evaluate the algorithm(s) on several problem domains to see how widely applicable the idea is. In this process, you may want to explore the effect of different hyper-parameters, such as the architectural choices, on the performance of the algorithm, and study issues such as sensitivity of the performance on the hyper-parameter.

When you perform an empirical study, you need to follow good statistical practices. For example, if there is any randomness in the algorithm (e.g., random initial weights of neural network), you need to run the algorithm multiple times in order to compute the mean performance as well as its high-probability confidence interval.

You also need to be careful in making sure that you separate your training/validation/test sets properly. Otherwise, your results would not be meaningful.

[2] You may also consult the following good set of suggestions by Csaba Szepesvári at https://rltheory.github.io/pages/assignments/, especially if you want to work on theoretical work. In addition to that, we consulted Animesh Garg's course on 3D and Geometric Deep Learning http://www.pair.toronto.edu/csc2547-w21/project/, and Roger Grosse's project instructions https://www.cs.toronto.edu/~rgrosse/courses/csc2541_2021/assignments/project.pdf for his Neural Net Training Dynamics. You may find good advice there too. Note that their evaluation criteria and what is acceptable or not is not the same as this course, so do not rely on that.

2.3 Algorithm and Architecture Design

You may decide to design a new algorithm or architecture by varying certain components of an already existing algorithm or architecture, and effectively search in the space of "adjacent possible".

The space of all possible algorithms and NN architectures is combinatorially large. Maybe we can use a computer to automatically explore it. For this course, however, we want you to explore it based on the insight gained in this course as well as your other courses and research experience.

You do not want to randomly wander in the space of all algorithms. Any new algorithm and architecture should be justified. You do not need to rigorously and mathematically prove that the algorithm works before trying it empirically, but it is good to have a theoretical insight before your start implementing.

2.4 Theoretical Analysis

Although this course is not focused on the theoretical analysis of NN/DL algorithms, if you are theoretically inclined, you may still decide to work on the theoretical understanding of an NN/DL algorithm or problem.

2.5 Special Topic: Application of DL on Sequences

If despite reading the previous subsections, you still feel lost in how to choose a project and require a bit more guidance, do not worry! We have decided to give a few more instructions and directions in a particular sub-area of NN/DL. This may give you more ideas about how to choose a project. Notice that there is no preference from our perspective in whether you choose working on this subarea or anything else that we described before. All are good, as long as you do a good job.[3]

In project type, your developed system must either take a sequence (of variable length) as an input, or produce a sequence as an output, or both. Your model should have an RNN or a Transformer component. If you want to use architectures that we have not covered in the course (such as diffusion models, neural ODEs, etc.), you are free to do so as long as you confirm your methodology with the instructors or TAs before submitting the project proposal. Here are some examples of possible projects:

- Using an RNN or Transformer to classify sequences. For example, decide whether a restaurant review is positive or negative.
- Using a generative RNN to produce sequences. For example, you generate Simpsons' TV scripts.
- Using a Siamese network to determine whether two StackOverflow questions are duplicates.
- Predict the next item in a sequence of, for example, stock values.
- Predict the outcome of a patient based on some sequential factors, such as their SO2 level and blood pressure.
- Predict the dynamics of objects under contact and collision, for example, in robotics or graphics applications.
- Generate molecules, or predict properties of molecules.

You are encouraged to use transfer learning and data augmentation ideas in your project.

3 Proposal

Your proposal should be a maximum of two (2) page summary of your intended research direction (references excluded in the page limit). You need to clearly

- define the problem (including brief motivation of solving this problem, and relevant dataset)

[3]This project direction is inspired from Florian Shkurti and Igor Gilitschenski's offering of the same course in Fall 2023. The following few paragraphs are copied from their project description almost verbatim.

- provide a brief summary of prior work, and
- what you intend to do (including some rough detail of DL algorithm/architecture you are considering),
- estimate compute resources that you need for this project and how you secure it.

The instruction team will provide you with feedback on your proposal. The more detail you provide, the better feedback we can give you. We also provide some office hours before the proposal deadline, in case you want to bounce ideas back and forth before submitting them in the written form.

4 Report and Source Code

You should write a 6-8 page report summarizing your work. We encourage you to use LaTeX.[4] You can use the NeurIPS style file https://nips.cc/Conferences/2023/PaperInformation/StyleFiles, though you are not required to use this specific style.

At a high-level, your report should include

- Problem definition and motivation: Clearly state what problem you are tackling and why we should care about it.
- Summary of prior work: What other attempts have been done in order to address this problem.
- Your contributions: How to describe it slightly varies depending on the type of project that you have. For example,
  - For an Application-type research, you should explain how you formulated your application as an ML problem; the description of algorithms and architectures you have tried; the experiment design to evaluate the performance of your algorithms, and the empirical results you obtained in comparison with other baselines.
  - If you are proposing a new architecture or algorithm, you have to clearly describe it and provide motivation and intuition for why you proposed architecture is a reasonable one. In addition, you need to carefully compare it with similar architectures or algorithms on some standard benchmarks.
- Conclusions: What have you learned and what is remained to be done or figured out?

Your report will be evaluated based on its quality of writing and explanations, how well you cover the prior work, precision of your statements, your contributions (which depends on the type of research you have conducted), and following the good scientific methodology.

It is common in research that each co-authors contribute to different aspects of the project. Some may come up with the high-level ideas, some design the algorithm, some study the idea theoretically, some design and conduct the experiments, and some others write the paper. We'd like to acknowledge that this is how modern science works and let you have different contributions. That being said, *you need to have a section describing the rule of each team member in the whole project in some detail*. The only requirement is that *all team members must be involved in writing the paper*. If you are not good in writing yet, the university is the right place to practice.[5]

We (tentatively) grade your report as follows:

- Problem Definition: 10%
- Prior Work: 10%
- Contributions (Method): 20%
- Contributions (Empirical Studies): 20%
- Clarity and Quality of Writing: 20%
- Conclusion: 10%

[4] If you do not know how to use LaTeX, this is a great opportunity to learn.

[5] If there is any reason that you cannot participate in writing the paper, for example a medical reason, you should discuss it with us beforehand.

- Description of Individual Contribution: 10%

Of course, if the project is theoretical, we adjust the Contributions (Empirical Studies) accordingly.

The source code of the project should be submitted. This can be in the form of a Zip file or a link to a public GitHub repository. It should be in working condition, which means that our TAs should be able to run it easily without any hiccups.

5 FAQ

Q: Is it acceptable to have a project that overlaps with my thesis project (CSC494/495 or similar)?

A: Yes, but you should be clear about what part has been done before this project, and what contributions are new. The basic idea is that you should not reuse your prior work for this project; you have to spend a significant amount of time during this semester to work on this project, but you can use it for your thesis (of course, if your supervisor is OK with it).

Q: Is it acceptable to have a project that overlaps with another course project?

A: Try to avoid it! If there is a good reason to have a project that spans more than one course, that can be discussed. You need to get the permission of all instructors for this. Which means that you need to send an email to us and the instructor(s) of the other course and get a joint permission. Since your research is done within a team and you may have different teams in different courses, this makes the credit assignment complicated, hence the discouragement.

Q: Can I extend the project from a previous course?

A: Yes! You should mention it in your proposal, include the report from the previous project in your submission, and be explicit about the new contributions specific to this course. In other words, be clear about the Δ.

Q: I do not know anyone else in the class. How can I find a teammate?

A: Use this as an opportunity to know your peers. You can stay around after the class or post a message on Piazza in order to find collaborators.

Q: Can I have a team size of 5+ or 1 or 2?

A: Teams should consist of three or four members, with four being preferred. Any other arrangement requires a clear reason and our prior permission.

Q: What if I discover that someone else has done a very similar thing to what I am doing in this project?

A: That is completely OK. It is a part of science. Make sure to cite those paper(s) in your prior work, and provide a detailed comparison.

Q: What should I do if the empirical results are taking too long to be ready?

A: You should be careful about the compute resources needed for the project, and that is one reason we ask you to contemplate and write about it in your proposal. Our goal is not that you work with a very large NN that requires several days or even weeks of training. Our goal is for you to experience research in NN/DL with as small NN as meaningfully possible.

As a rule-of-thumb, if it takes more than half a day to train your NN and get reasonable initial results, your NN and dataset are too large. You need to consider that you probably need to go through several architecture iterations and debugging rounds before eliminating all the bugs and finding a good architecture.