Machine Learning projects

Mathematical Sciences for AI B.Sc. @ Sapienza University of Rome

A.Y. 2023/24

Prof. Emanuele Rodolà, rodola@di.uniroma1.it
Dr. Adrian Minut, robertadrian.minut@uniroma1.it
Dr. Daniele Solombrino, solombrino.1743111@studenti.uniroma1.it

Introduction

In this document, we present the projects for the Machine Learning exam in A.Y. 2023/24 🎇

Every project covers different aspects of the course, so make sure to select the one that catches the most of your interests.

Originality and creativity is appreciated and rewarded, so, if you want to go further than the minimum requirements listed for each project, you are more than welcome to do it!

Grading

As per course rules, this project accounts for 70% of your final grade and can be submitted in any of the available calls of the exam, independently of when and whether you already did the theoretical exam.

Needless to say, you need both the theoretical and practical parts to register the exam 🙂

Make sure to submit your project at least 7 days before the Infostud date for whatever exam call you choose; otherwise, we cannot guarantee a timely mark registration and you'll have to book to the next available call.

The project will be graded according to the following aspects:

I. Narration
II. Experimental results
III. Methodology
IV. Usability

Each aspect will be graded in the following scale: 18 / 22 / 26 / 30. The final project grade is the mean of the votes. The *cum laude* is up to the teacher.

Code

Taking code from external sources is more than fine, and a welcome practice in academia and industry, provided that the **original source** is **cited**. Submitted code must be **reproducible**, as we may want to dive deeper into it for an extensive review.

This is another important practice in academia and industry, which unfortunately is often ignored, wasting time and resources on both the producer and the consumer end 😡

Submission

To submit your project, send an email to the three addresses at the beginning of the document with the following subject: **[ML 2023/24] project submission**.

- The project must be submitted as a Python **notebook**.
- If you require extra files (e.g. to store your classes, functions, libraries) you are free to use them; but our entry point to your project must be a notebook.
- Also submit a requirements.txt containing the necessary libraries to run your code.
  If the notebook does not execute correctly because of missing libraries or other issues, we won't spend time trying to fix it.
- The notebook must include clear explanations, reasoning, code, results, plots, discussion, illustrations… There are no rules, but the clearer the notebook, the higher the grade.

Specify the git repository hosting your project in the body of your email.

How to contact us

In case of need, please reach us at the three addresses listed above, using the subject **[ML 2023/24] project assistance**.

For bureaucratic or sensitive reasons, you can message just Prof. Rodolà.

A list of projects follows below.

1. Report on an open source project

Being able to understand code written by someone else is a very important skill you need to have in you, especially in ML, as you may want to use existing code as a starting point.

In this project, you can take whatever machine learning project you can find online, make sure it is working as advertised (e.g. verify on your machine that their 99% train, validation and test accuracy claim is genuine), and then write a report about it.

If the code you picked is not shared as a Notebook, please convert it first to Notebook format and then write your report inside it.

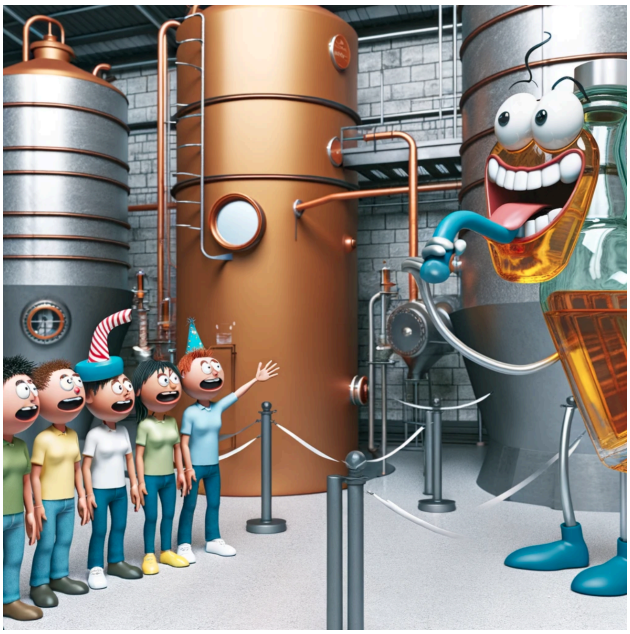
2. Multi-objective dimensionality reduction

During our lectures, we have seen some techniques related to dimensionality reduction, like PCA and t-SNE. Both methods optimize an objective function, which enforces a desired behavior. What happens when we try to enforce additional, non-trivial objectives during the optimization of such problems?

For this project, you will:

1. Select a dimensionality reduction technique.
2. Find an extra objective you would like your technique to exhibit (e.g. enforce sparsity in the lower-dimensional space learnt by t-SNE).
3. Formalize your extra objective and incorporate it into the original energy.
4. Motivate, discuss, experiment with, and champion your new technique.

3. XS knowledge distillation



Knowledge distillation is a deep learning technique where a smaller, simpler model (student) is trained to replicate the behavior of a larger, more complex model (teacher).
Different techniques across a wide range of datasets and tasks can be successfully applied to distill knowledge from a larger neural network to a smaller one.

The goal of this project is to experiment with knowledge distillation starting from very small teachers, like MLPs with just a bunch of layers, or non-neural models (like ensemble methods).

Considering the very experimental and unexplored nature of this project, you can choose whatever task, dataset and knowledge distillation technique you like the most (choosing the simplest knowledge distillation technique is more than fine for the scope of this project!)

Available resources:

- Knowledge distillation:
  - https://arxiv.org/abs/2304.04262
  - https://arxiv.org/abs/2006.05525
  - https://hanlab.mit.edu/courses/2023-fall-65940 (just knowledge distillation lectures)

4. Neural dimensionality reduction

During our lectures, we have seen some techniques related to dimensionality reduction, like PCA and t-SNE. Both methods optimize an objective function, which enforces a desired behavior, but none of them uses a neural network to reach its objective.

The goal of this project is to perform dimensionality reduction using a small neural network on a simple dataset (e.g. MNIST) and then compare the learnt embedding space to the one learnt by PCA or t-SNE.

Resources:

- Lecture on PCA.
- Lecture on t-SNE.
- Lecture on Metric Spaces.

5. Machine unlearning

Can you ***unlearn*** something?

Your task here is the following: given a learning model (for example, an MLP, or some ensemble model, or linear regression, your choice!) pre-trained on some data, you want to modify it to selectively forget a class, and learn a new class.

Here's one possible approach which uses a MLP, and you may want to start from here (but not necessarily -- again, it's your choice). Start with a MNIST classifier pre-trained on a subset of the digits. Now replace one of the learned digits, say the class "6", with a new digit, say "3". A possible way to proceed is to identify which weights are more involved in the prediction of class "6", freeze all the rest, and train with a loss that favors the "3" while penalizing the "6". Test this baseline and see whether it brings you anywhere. Are there any pitfalls in this idea? Does it work? Use it as a first line of attack to understand the problem.

Starting from these baseline tests, devise a new unlearning procedure. You can improve upon this baseline, make up your own idea from scratch, or check the literature to get ideas. If you use an existing approach, ***you must add something new***, for example by testing it on some new data modality (e.g., audio), by studying more extreme cases, failures, weaknesses, or by making it more efficient, and so on.

6. Transfer Learning on shallow models

Transfer Learning is a very powerful technique which uses knowledge from a model trained for another task or on another dataset as a starting point, when training a neural network for another task or on another dataset.

The goal of this project is to experiment with transfer learning on shallow models, like ensemble models.

Given the experimental nature of this project, you can choose whatever setting (i.e. task, model and dataset) you like the most.

The only requirement is to use *non-neural* models.

7. AWOL - Amusing Wails On Loop

In this interesting paper (https://arxiv.org/abs/2404.03042), the authors were able to show 3D geometry generation ***using language as a bridge***. Can we do the same for ***generating audio*** (speech, individual instruments, music, you name it)?

In this project, you will take the simple pipeline from the AWOL paper linked above and adapt it to the audio setting. According to the authors, the proposed pipeline is very easy to train (down to 5 minutes), requires very few training examples (as little as just one), and generalizes very well out of distribution. Can you achieve this with audio?

You have freedom in choosing the type of data, the external synthesizer / music generation software to use as a parametric model, the final application, and so forth. Chances are that whatever you do will be cool and novel, and it will be fun for the very reason that you are going to generate new sounds! We believe that even noise is fun to hear (so much so that this is even a musical genre), while noisy images are arguably as fun.

Of course, this doesn't mean that a project that doesn't work will get a good grade :)

8. Bot Playing "Dixit" Board Game

"Dixit" is a creative board game where players describe images in imaginative ways. An agent capable of playing "Dixit" may be created by combining a CLIP-like model for image understanding with a large language model (LLM) for generating and interpreting creative descriptions. We want an agent capable of playing both roles in Dixit: who guesses and who gives the hint.

In this project, you will:

- Develop a bot to play Dixit, based on a pre-trained CLIP-like model to interpret images and a LLM to generate descriptions (GPT APIs are fine).
- Conduct experiments to compare the bot's performance against GPT-4o multimodal and humans in >10 games.

Hashtags: #generative-AI, #multimodal-models

References:
"Creative Captioning: An AI Grand Challenge Based on the Dixit Board Game" (https://arxiv.org/pdf/2010.00048)
Dixit images: (https://github.com/jminuscula/dixit-online/tree/master/cards)

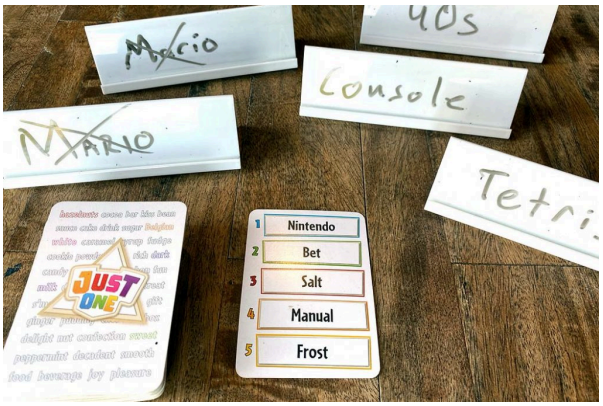
For this project, refer to Dr. Antonio Norelli - Research Associate in CS at the University of Oxford <antonio.norelli@cs.ox.ac.uk>

9. Coordination of LLM agents in Just One

"Just One" is a cooperative board game where players give single-word clues to help others guess a mystery word. This project aims to create a multi-agent system where different bots collaborate to play "Just One," each one using a LLM to generate and interpret clues. This is a toy problem that aligns with recent and promising research on LLM agents.

In this project, you will:

- Develop collaborative bots based on LLMs to play Just One. Your bots should be able to generate single-word clues and guess the mystery word depending on their role, following the rules of Just One.
- Compare the bots' performance with human game statistics from Board Game Arena for Just One (BGA).

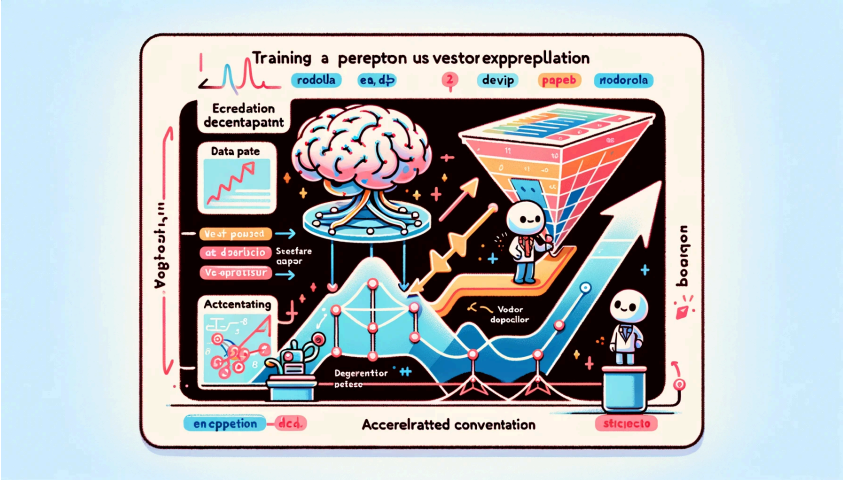


Hashtags: #generative-AI, #LLM-agents

References:
LLMs agents: (https://x.com/AndrewYNg/status/1780991671855161506)
Just one description: (https://boardgamegeek.com/thread/2348677/just-one-a-detailed-review)

For this project, refer to Dr. Antonio Norelli - Research Associate in CS at the University of Oxford <antonio.norelli@cs.ox.ac.uk>

10. Train a MLP using vector extrapolation



Have a look at this paper (and those cited within, but reading this paper is already good enough): https://cvg.cit.tum.de/_media/spezial/bib/rodola-bmvc13.pdf

Many years ago, we had this idea of using a simple technique, based on solving a simple linear system, to accelerate the convergence of gradient descent methods. We applied this to the shape matching problem, and it worked beautifully!

Now the question is: *Can we use the same exact idea to accelerate convergence of neural network training?*

For this project, your task is to take a vector extrapolation technique, such as those included in the paper above or others you might find elsewhere, and apply it to training neural networks. The hope is that you can achieve faster convergence! There are a few hyperparameters to tune, and everything might become a bit too costly; but you know what to do, apply random search, train with small data and then scale up, in other words: apply the tricks of the trade to get to your objective.

Remember that ***even a negative result is a good result***. If nothing works, explain why, give possible explanations, show your experiments, and draw conclusions!

11. Parameter dimensionality reduction

During our lectures, we have seen some techniques related to dimensionality reduction, like PCA, t-SNE and SVD, and witnessed their excellent results in creating a compact representation for input data.

The goal of this project is to understand the impact on performance of applying dimensionality reduction to the layers of a trained MLP at inference time.

Considering the experimental nature of this project, you can choose whatever task, dataset, and dimensionality reduction you prefer and you can also consider algorithms that we did not cover during the lecture.

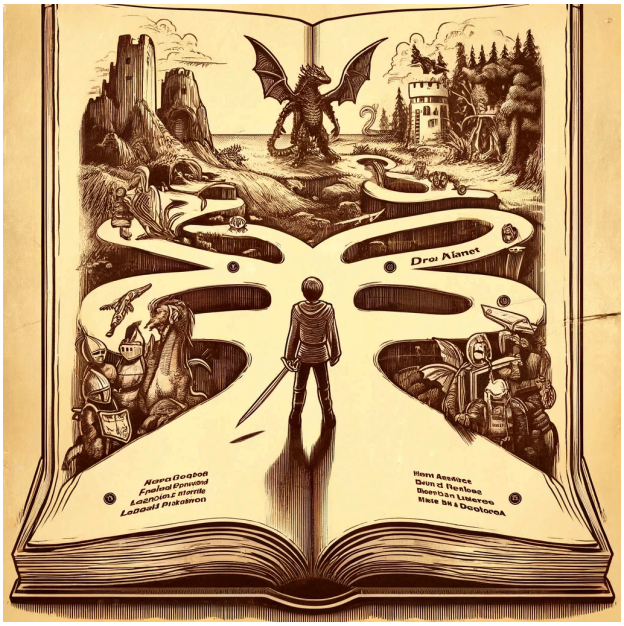
12. Kaggle

Take part in one of the open Kaggle challenges!

For this project, you choose your own competition among the ones listed on https://www.kaggle.com/competitions. The rules are as follows:

- You **can't** select one of the competitions that run indefinitely with a rolling leaderboard. More in general, you can't choose a finished competition where the winning solutions have been made publicly available, or where the test set is publicly available.
- Your project will be graded only if you show your current standing in the competition's **leaderboard**. You don't have to be a medalist; you can even be the last in the entire leaderboard, but you must be there!
- **It is ok** if your solution makes use of hints and discussions from the Kaggle discussion board.

13. Choose your own adventure



If none of the above projects catches your interest, that's totally fine!

We are open to suggestions, especially if centered on the topics proposed in all other tracks.

Feel free to email us using the subject **[ML 2023/24] custom project proposal**.

Note: We will not accept every project proposal; in particular, the proposed project ***can not be recycled from other courses***, and it must be related with ML. If you don't get our explicit approval on your proposal, it won't be considered valid for the exam.