COMP6721 Applied Artificial Intelligence
Course Project Guideline
Posting Date: Friday May 12th, 2023

Table of Contents

Resumé

Dataset	No. of Observations	No. of Features	Goal
You choose	> 20k	>= 15 (Mix of numerical, categorical, images, or text)	Classification

Models	Steps
Decision Tree supervised Decision Tree semi-supervised DNN supervised (CNN, RNN, …)	Data preprocessing and cleaning (Feature selection and engineering) Visualisation Modeling Report

Section	Deadline (11:59PM)	Deliverable	Evaluation
Team Formation (4p)	Tuesday, May 16th	Email to Lecturer	-
Confirming the Dataset with TAs	16th - 21st POD sessions		-
One-page proposal	Sunday, May 21st	PDF file in Moodle (1+1 page)	10%
Progress Report	Tuesday, June 6th	PDF file in Moodle (2 pages)	20%
Final Report (6 pages) README.txt Sample test dataset One-page contribution Deck of 8 Slides in Moodle	Thursday, June 17th	ZIP file in Moodle	40%
Final Presentation	Thursday, June 17th	Recorded video (e.g., with screen capture and camera in PowerPoint) Deck of 8 Slides in Moodle	15%
Code submission	Thursday, June 17th	GitHub	15%

Email Inquiries

Lecturer:	Arash Azarfar	[email: arash.azarfar@concordia.ca]
TA/POD	Soorena Salari	[email: soorena.salari@mail.concordia.ca]
TA/POD	Farzad Salajegheh	[email: farzad.salajegheh@concordia.ca]
TA/POD	Y A Joarder	[email: ya.joarder@concordia.ca]

All inquiries about the course project should be communicated via *email* using the addresses above. Your email subject line must follow a prefix topic.
[COMP6721: *{your subject}*].
For example, if you are inquiring about the team formation, the subject line would be

[COMP6721: Course Project-Team Formation].

Main Objective

The main goal of this project is to study a Machine Learning (ML) task using both Decision Trees and Deep Neural Networks to address a machine learning problem from a real-world subject of interest, applying supervised and semi-supervised learning Classification. Each team is at liberty to choose any dataset, image or text, under the following circumstances:

1. Availability of datasets online: you will select the same dataset for all the steps of the project. With the same dataset, you will implement the following learning tasks (in addition to required data preparation and visualization steps) and compare the performance.

I. Supervised learning Classification with Decision Trees
II. Semi-supervised learning Classification with Decision Trees
III. Supervised learning Classification with a deep learning model (CNN, RNN, …)

2. The datasets should have at least 20K observations. You have the freedom of eliminating parts of the datasets if you need to, given that you still meet the requirements. Note that you cannot combine datasets from different sources together.

3. There is no right or wrong choice of problem. Choose a problem that interests you, given it satisfies the availability of varying datasets. You could work on image or text datasets. You could also suggest other types of dataset and confirm it with your TA.

Once a problem is chosen, each team needs to train, optimize, and evaluate a Decision Tree and a DNN architecture to tackle the chosen classification problem. For the semi-supervised part, you will train multiple Decision Trees iteratively taking into account the labeled data and more confident predicted labels. All deep neural network (DNN) methods must be implemented using the PyTorch library. Specifically, the project has the following 3 main components:

I. Descriptive and exploratory data analysis (EDA)
II. Data Preprocessing: In this stage, the team needs to explore several data preprocessing methods.
III. Optimization: Each team attempts to optimize their model through hyperparameter tuning. You should choose at least three hyper parameters for the tree (e.g., depth, number of branches, pruning option, etc) and two for the neural network (learning rate, batch size, activation function, and loss function.)

Please note the following

1. You should consider the computational complexity of the selected network models that need to be trained on a commodity GPU hardware from available resources (e.g., lab GPU, cloud, personal PC, Google Colab, etc).

2. You should use several evaluation metrics, in addition to plots, to compare the performance of the different models. This includes, but not limited to, accuracy, recall, precision, F-score, etc. Each team should also use TSNE or Grad-CAM to visualize the performance of different models in the final report.

Team Formation **[Deadline: Tuesday 11:59PM, May 16th, 2023]**

Students are required to form a team of *Four (4) members* for the course project. Please engage in discussion with your classmates for brainstorming on potential the problem you would like to solve according to the main objective above. Please submit your team's details *by email* to the **lecturer** following the email inquiries guideline. A *Q&A discussion forum* is created for the course on Moodle, and you can use the platform to open a discussion on team formation related topic. Students who cannot find a team will be randomly shuffled into incomplete teams. The team, once formed, will stay the same until the end of the semester.

Writing a One-Page Proposal

You should write a one-page proposal for the course project to cover the following topics:
- *Problem Statement and Application:* Provide a background about the topic and specify why the problem is important. What are the associated challenges of your selected problem application? What are your expectations/goals throughout developing the application of interest?

- *Dataset Selection:* Explain your dataset and provide the statistical details of your data (e.g., number of observations, type of features, etc). Specify where you have found the dataset and means of access to the data (e.g., published paper, downlink, etc). You may consult Kaggle ,UCI, or Google Dataset Search to find possible datasets of interest, or you can use any data you have collected from your previous research activity or have access to.

- *Possible Methodology:* Highlight the "possible methods" you could use to solve the problem. Specify how you will be handling/processing the data to train your deep learning pipeline using a CNN or RNN model, for instance. Furthermore, discuss the metrics that will be used to assess and evaluate the pipeline, and your expectations regarding the kind of results/performance to be achieved. You need to discuss the possible method(s) and how the obtained results will be compared and analyzed to each other. Further, you need to discuss the potential of your analysis and comparisons and how they can be useful for scientists and engineering in the field.

- *Bibliography:* You can add an additional page (if needed) to extend your reference list cited in your proposal. The citations may include, but not limited to, published papers and domain links. (include a link to your dataset). Please note that failure to properly cite your references constitutes plagiarism and will be deemed for reporting.

You will be given the opportunity to submit your proposal or discuss your dataset selection for revision by the TAs, before the final graded submission.

Proposal Submission **[Deadline: Sunday 11:59PM, May 21st , 2023] (Counts for 10% of the course project grade)**

Only the admin (one person) of your team needs to upload the proposal in *PDF* file in Moodle. For the report format, please consult "Reports Formatting" Section in the third page. Our team (TAs and lecturer) will review your proposal and, if it is acceptable, you may proceed with

developing the next phase of your project. Otherwise, we will instruct you to either revise or re-write the proposal according to the guidelines of the course project. All teams are highly encouraged to put great effort into preparing the first proposal draft to avoid further delays in project developments.

Progress Reporting **[Deadline: Tuesday 11:59PM, June 6th, 2023] (Counts for 20% of the course project grade)**

Each team is required to submit a two (2)-page progress report highlighting the main steps taken after the proposal, and any initial results from the applied models (if available). The progress report should contain the following sections:

1. *Introduction and problem statement:* In addition to defining the problem and its applications, discuss the general strategy for tackling the issue at hand. Discuss the challenges faced in solving this problem and any possible solutions to address them. Discuss what results you expect and how you want to acquire/evaluate them.

2. *Proposed Methodologies:* Give updates regarding the methods used/to be used. Discuss the chosen dataset and the model in more detail than the proposal.
3. *Attempts at solving the problem:* elaborate on failed or successful attempts at tackling the problem. Furthermore, discuss any possible/preliminary results.
4. *Future Improvements:* Discuss briefly how and where you want to change to improve the accuracy of the model.
5. *References:* You can add an additional page (if needed) to extend your reference list cited in your progress report. The citations may include, but not limited to, published papers and domain links (include a link to your dataset). Please note that failure to properly cite your references constitutes plagiarism and will be deemed for reporting.
6. *Supplementary Material* [this section is appended to the main report draft]:
You may include appendices to your report to support different sections of the main draft.
**Note: this section will not be considered for marking. Furthermore, reviewing this section for the lecturer and TAs is not mandatory.

The progress report should be in PDF format and uploaded in Moodle. For the report format, please consult "Reports Formatting" Section in the third page. Please note only the admin (one person) of your team needs to upload the progress report in PDF file in Moodle.

Final Reporting **[Deadline: Saturday 11:59PM, June 17th, 2023] (Counts for 40% of the course project grade)**

The final report should articulate the following sections:
1. *Abstract.* Articulate the abstract presentation of the project and what to expect by reading your report in full detail. Briefly discuss the problem, proposed methods and used data, and the achieved results. [maximum of 150 words]

2. *Introduction* [the abstract & introduction should be no longer than 1.5 pages].
a) Write a section to cover the problem statement and its importance to the application field. What are the associated challenges with respect to the problem? How these challenges have been

addressed in the literature? What are the pros/cons of the existing solutions? How is this report trying to solve the problem and a challenge in mind? Elaborate on the high-level abstract explanation of your methodology and what kind of implementations you have done. What kind of results you are obtaining?
b) Related works. Write a subsection to cover literature review and related work descriptions.

3. *Methodology* [this section should be no longer than 2 pages].

The methodology section should cover several subsections as follows:

a) *Datasets*. A comprehensive description of the datasets, including where and (how) there were collected, a complete statistical details, distribution and analysis of the datasets such as size of the data, number of observations, number of classes, and any preprocessing and filtering steps you have taken to make it ready to be fed to your models. Explain your train/validation/test breakdown, cross-fold validations, resolution level for training, etc

b) *Decision Tree Model*. Describe the architecture of the decision tree.

c) *DNN Model*. Describe the architecture of the selected DNN models Elaborate on why you think the selected models are suitable for your practice. Describe the computational complexities of the selected models for training and validation phases in terms of wall clock time for one-epoch training as well as number of FLOPS calculation.

c) *Optimization Algorithm*. Discuss how you validated and optimized your model. What optimization algorithm(s) you are choosing to train the DNN model? What metric evaluations are considered for reporting the performance of the optimization algorithm. Describe the properties of the algorithm and its associated hyper-parameters for training.

4. *Results* [this section should be no longer than 2.5 pages].

This section describes and analyzes the experimental design and obtained results in detail. More specifically

a) *Experiment Setup.* you need to describe how you setup your experiments, optimized and validated your models, the performance of your models using appropriate metrics (precision, recall, F1-measure, …). Explain the ranges of hyper-parameters and rational behind selecting as such in relation to your data and models.

b) *Main Results.* Demonstrate the main results in figure/table formatting and analyze the performance of your trained models, as well as comparison with other available results.

c) *Ablative Study*. Demonstrate the ablation results from tweaking different hyper-parameters such as number of classes for training, number of images per class training, different range of learning rates, different range of batch-size, tree depth, branching, pruning, etc, and explain your observations.

5. *References* [this section lists all references on your report]:
Cite any references you used in the projects, including any source code and dataset you have used in the project. Please note that failure to properly cite your references constitutes plagiarism and will be deemed for reporting.

7. *Supplementary Material* [this section is appended to the main report draft]:
You may include appendices to your final report to support different sections of the main draft. **Note: this section will not be considered for marking. Furthermore, reviewing this section for the lecturer and TAs are not mandatory.

Reports Formatting

The proposal (1 page + 1 page bibliography), the progress report (2 pages+1 page bibliography), as well as the final report (6 pages + possible appendices) should **all** be written in IEEE 2-column conference format and submitted as PDF (You may use Word or LaTeX). Note to use the *reviewing style* for LaTeX compilation.

https://template-selector.ieee.org/secure/templateSelector/publicationType

https://cvpr2022.thecvf.com/sites/default/files/2021-10/cvpr2022-author_kit-v1_1-1.zip

Final Presentation **[Deadline: Saturday 11:59PM, June 17th, 2023] (Counts for 15% of the course project grade)**

Each team should prepare a six (6) minute recorded video from a slide presentation and submit the following
- A 8–10-page deck of slides prepared in PDF format (you can use either PowerPoint or LaTeX beamer for your slide preparation). Slides should contain a high-level overview of the problem and goals, the type of data you were dealing with, your methodology, the obtained results, and the references used.
- A six (6)-minute recorded video from the team, each member taking a round of 1-2 minutes in a row to complete the record.

GitHub Submission **[Deadline: Saturday 11:59PM, June 17th, 2023] (Counts for 15% of the course project grade)**

Whether you use Git to organize your coding throughout the project or not, each team should create a new GitHub page for the project from the beginning. The GitHub page should be created in "private" mode and each member should be given access to commit their updates on a regular basis during the course of project. Furthermore, the assigned TA for the project team as well as the lecturer should be given access to the GitHub page for monitoring the progress of the team. Note that git commits from each team member will be monitored for the engagement of individuals and considered as one of the means of marking to contribute to their final project.

The final GitHub page should contain the following:
- High level description/presentation of the project
- Requirements to run your Python code (libraries, etc)
- Instruction on how to train/validate your model
- Instructions on how to run the pre-trained model on the provided sample test dataset
- Your source code package in Scikit-learn and PyTorch
- Description on how to obtain the Dataset from an available download link

Please note that if the instructions to run your code are incomplete or not explicit enough, you might lose marks for that part of the project. You should add the professor and the TAs as contributors to your project.

How to Submit Your Project Materials? **[Deadline: Saturday 11:59PM, June 17th, 2023]**

Submit all the files in one zip file including
- PDF file of the final report
- Deck of 8-10 slides presentation in PDF format
- README.txt containing the following two links
  - A link to your GitHub page.
  - A download link to your video presentation

- A sample test dataset
- **One page that includes a table listing the contribution of each team member to the project**. The table format should be in Four (4) columns pertinent to individual members of the team. The pertinent information will be considered to grade individual contribution to the project.

The zip file should be uploaded by the admin of the team in Moodle by the final submission deadline.

Late Submissions

If you submit any part of the project later than the specified deadline on Moodle, your submission will be accepted until the cut-off date. However, you will lose 20% of the mark for each day you submit late. The cut-off date is maxed up to two (2) days and submission after the cut-off date will not be accepted. Further, please note that resubmitting your files will result in erasing all the previously submitted versions and their respective dates. The date of the last attempt at submission will be counted as the final submission date.

APPENDIX 1 Semi-Supervised Learning

In an ideal word, data are labelled, meaning we have a large dataset with the value of target variables (e.g., image subject, object, text sentiment, etc) already determined for each observation. However, in the real world, it is usually very costly to label the data, so perfectly labelled datasets are rare! Consider the example of image recognition or image context prediction. It is easy to do a

web search and find thousands of images (our features), but the data will be unlabeled. Companies may hire employees whose tasks are just browsing the images and assigning the right label.

Semi-supervised learning is a machine learning method (we may say a sub-method of supervised learning) which targets these situations where we have a large dataset, and the majority of the observations are unlabeled. However, a low percentage of the observations are labelled (usually less than 20%).

The main idea in semi-supervised learning is as follows:

- We do supervised learning with the labelled data only (let's say 20%).
- The resulting model is applied to the unlabeled data, and unlabeled observations are pseudo-labelled (the rest 80%).
- Among the observations with pseudo-labels, we take the ones with a high confidence (e.g., the predicted probability instead of labels in Decision Tree, Logistic regression, or Naïve Bayes models) (Let's say top 10% predictions)
- These high-confidence observations are mixed with the originally labelled data to form the new labelled subset (now 20%+10% = 30%). Ignore the predicted pseudo labels for the rest.
- We re-run the steps above (We have now 30% "labelled" and 70% unlabeled).
- Usually, 5-10 iterations are required to finalize the labels (predictions) for all unlabeled data.

So, for your dataset,

- Put aside the same final test set (10-15%) that you used for the supervised learning (for a faire comparison).
- Among the others, randomly select 20% of data as labelled and ignore the labels of the 80% (consider them unlabeled).
- Make your supervised learning model and predict (pseudo-label) these 80% observations.
- Check the probabilities and select the ones with high confidence (e.g., =<0.15 and >=0.85 for a binary classification). Mix them with the labelled data and reiterate.