CS 595a Final Project Instructions

Spring 2023

Table of Contents

1 Overview

The final project for *CS 595a: Artificial Intelligence* is to select a problem that interests you that you believe can be addressed by artificial intelligence and/or machine learning, investigate and implement a solution to the problem, and provide analysis of your resulting system toward relevant performance metrics.

This is a group project (with a group of three students enrolled in CS 595a). Unfortunately, due to differences in deliverables, we cannot have teams of students mixed across courses.

2 Project Selection

Potential project topic:

- **Investigation and exploitation of a data set**
    - e.g. develop a classifier, predict a component failure, estimate a cost given historic data, image classification, etc.
    - A variety of data set can be found online. The following are some of the resources to consider:
        - AI Crowd: https://www.aicrowd.com/
        - Kaggle: http://www.kaggle.com
        - Google Public Datasets: https://cloud.google.com/public-datasets/
        - NASA Data Portal: https://data.nasa.gov/
- Above suggestions help inspire you, but you are free to propose other AI problems, and are not limited to only topics covered within this course.

The remainder of this document goes through the requirements for each of the project deliverables.

3 Deliverable #1: Proposal [10%]

The proposal should be 1-1.5 pages in length. Its purpose is to allow you to organize and present the project that you wish to perform. You will be given the opportunity to briefly describe the problem(s) being addressed, the approach taken, any resources to be used (e.g. data sets, libraries, test environments, and such), a work breakdown structure, list of team members, and assignment of major tasks to team members.

What you propose is not set in stone and it is expected that your approach, resources, etc. will be subject to change as you proceed with your project.

Proposal Rubric: The following rubric describes the sections of your proposal and the expectations for each. Additionally, professional technical writing is expected (typo free, well-formatted, grammatically correct).

Section	Expectations	Points
Problem Statement (200 words max)	• Clearly describes the problem being addressed • References any external resources that support this problem (e.g. data sets, simulation/game environment, etc.)	20
Proposed Approach (300 words max)	• Identifies the type of AI/ML approach to be used to address the problem, e.g. o Search o Classification o Prediction/regression o Reinforcement Learning • Describe how your problem relates to the proposed approach (i.e. how do you map the problem to the approach?) • Identify any planned libraries, resources, etc. you intend to utilize to facilitate your approach	45
Work breakdown structure	• Breaks down the project into a general timeline of major milestones, the date to be completed, and the team member(s) assigned to the task. • Note: be sure to include tasks to address the requirements below	15
Team Structure	• Identifies team members • Identifies any relevant background of team member(s) to the problem, if applicable	15
References	• Document includes references to any frameworks, simulation environments, data sets, problem descriptions, etc. used in this write-up • Citations and inline references are in IEEE format	5

4 Deliverable #2: Presentation and Demonstration [25%]

Each project will be presented during the final week of class (Monday or Wednesday). A presentation time slot will be assigned at least one week prior. The presentation and demonstration can be presented live, or it can be a recorded video presented to the class.

The amount of time per presentation will be dependent upon how many projects are proposed. For now, anticipate 5-8 minutes per project max.

Your presentation and demonstration must include:

- Overview of problem
- Summary of approach taken
- Demonstration (one of the following, as appropriate to your project)
  - Live demo,
  - Video of software in action (a "highlight reel"), or
  - Presentation of Jupyter notebook with data analysis and ML results
- Conclusion including key lessons learned

The rubric for your presentation is as follows:

Section	Expectations	Points
Overview of problem	• Presentation clearly summarizes the problem that the project addresses. • Identifies the problem's scope including any limitations and delineations	20
Summary of Approach	• Identifies algorithm(s) used • Identifies key resource used • Summaries how the algorithms and resources were combined to produce a solution to the problem.	20
Demonstration	• Demonstrates major achievement of the project toward the problem • Demonstration should be easy to follow for the audience • Demonstration should be planned and rehearsed.	20
Visual Quality	• Suitable visuals are used for the presentation and demonstration • If slides or a Jupyter notebook is used to guide the presentation, all text and graphics should be readable by the audience from anywhere in the classroom	20

	• Visuals are well organized and serve a purpose within the presentation and demonstration.	
Verbal quality	• Presentation appears suitably practiced (student(s) know what to say and levels of umms are kept to a minimum) • Speaking is clear and professional	20

5 Deliverable #3: Source Code [40%]

Your team is responsible for submitting your source code through a public Github.com repository.

As students, you own the software that you develop and are free to include a license file stating whatever license you wish to place on your software. If your repository contains open source software appropriately used as per its license, you must include any requisite licensing documentation as applicable by the sources license.

You are not obligated to submit a project in Python. While Python is the language used for course instruction, there exist non-Python frameworks, simulation/game environments. You are free to use the resources that best meet the needs of your project.

Your repository should provide a read me with clear instructions on how to compile and execute your work. Build files such as makefiles, gradle files, script, etc.. are appreciated when applicable. It must also provide clear instructions on any required resources and instructions (or links to instructions) to install those resources.

You are free to use 3rd party libraries (SKLearn, Pandas, Tensorflow, Keras, etc.), simulation/game environments, cloud-services, etc. as necessary to complete your project so long as the final solution (i.e. connecting these resources to solve the AI problem) is your own.

Be sure to cite within your comments any use of external code regardless of the source's license.

The following rubric will be used to assess the source code:

Criteria	Expectations	Points
Completeness	• The software implements a solution that applies artificial intelligence/machine learning to solving the problem. • The repository contains a complete solution based upon the work described within your report. • Upon following all set-up instructions, the solution compiles and executes. • A README file is included meeting the requirements specified above.	35

	• If a complex system set-up is required, scripts are provided to guide the set-up.	
Correctness	• The software executes as described within the report. • The software correctly makes use of any external resources (libraries, APIs, etc.) • The software appears to be free of any obvious defects. • The software correctly utilizes the artificial intelligence algorithm's in tools in the implemented approach.	35
Style	• Each source file includes a comment block identifying the author(s), the project title, and a description of the file • Each class and method should have a comment block describing its function, parameters, and outputs (as applicable) • Proper variable naming is used such that non-iterator variables have names that convey the meaning/intent/purpose of the variable • Code makes use of whitespace to enhance the readability and interpretability of the software.	20
References	• Any external resources are documented in the README file. • Use of 3rd party developed source code (including code snippets borrowed from Stackoverflow) are properly cited in the code with a comment and/or other appropriate documentation as per the source's license.	10

6 Deliverable #4: Project Report [25%]

A report must be submitted with each project. The report should be technical in nature and showcase the work done. The report should be well organized and professionally written.

It shall include:

- Introduction (0.75 – 1 page in length)
  - Introduces the problem addressed by the project.
  - Summarizes the approach taken.

- Related Work (1-2 pages)
    - A literature survey should be conducted citing previous efforts to address the problem.
    - Journal articles and conference papers are preferred.
        - Blogs, online videos, etc. are acceptable, but should be used sparingly and only when suitable peer reviewed sources are not available.
        - Articles published in Arvix are also acceptable, but are preprint versions of the paper and may not be peer reviewed as published on their site.
- Approach (length as needed)
    - Data/problem analysis (if applicable):
        - Discuss any work done to analyze the problem further to guide your implementation.
            - For example, data set visualization and preliminary analysis
        - Survey of tools/resources available
    - Resources used:
        - Briefly describe any external resources used to support your solution
    - Software design
        - Show through both figures and accompanying text the design of your software solution
            - e.g. flow chart, software system block diagram, etc.
        - Tell the story of how you solved through implementing your own software, connecting various resources together, etc.
    - Source Code Description:
        - In a table, provide an overview of the files within your repository including filename and description.
- Evaluation and Results
    - Discuss and justify how you evaluated your solution to determine it is functioning as intended. For example, as applicable, this could include
        - Experimental methods
        - Metrics used
        - Comparison benchmarks used
    - Present results
        - Utilize figures that demonstrate how well your solution solved the problem
            - e.g. ROC or Precision-Recall curve for a binary classifier
        - All figures and results are also discussed within the text of this section.
    - Result Discussion
        - Discuss how your results reflect your ability to solve the problem.
        - If your results show issues with your solution, discuss how this is reflected in the results and what lessons learned can be pulled from the results.
            - Some projects will try things that do not work. If that happens to you, you need to describe how you know it did not work, and

provide some insight from your experience on how you could improve things for the future

- Conclusion
  - Briefly summarize your conclusions from the project including, but not limited to:
    - Lessons learned from your results
    - Lessons learned from the experience (technical lessons learned)
  - Describe next steps to be considered if someone were to extend your work
- References
  - You must use IEEE style inline citations, and IEEE style references for this document.
  - Your references section should reference:
    - All papers cited
    - All external resources discussed as part of your solution

The following rubric will be used to assess the final report:

Criteria	**Expectations**	**Points**
Introduction	• The introduction clearly describes the problem, the approach taken, and any external resources that were vital to the success of the project. • The introduction should also state if the approach taken was successful in addressing the problem, or not.	10
Related Work	• Summarizes related work relevant to the project. • Survey of related work is well organized permitting the reader to understand the relevance of the previous work to the current problem. • 1-2 pages in length • Primarily cites quality peer reviewed sources such as journal articles or conference papers.	20
Approach	• The author's describe how they analyzed the problem and/or data set prior to implementing a solution. • Section adequately conveys the solution to the problem as performed for the project. • Resources used are clearly described and justified. • A table is included that summarizes each submitted source code file.	20

	• Software design aides the reader in understanding the source code submitted, the software system’s assembly, and how the software flows.	
Evaluation and Results	• A method for evaluating the performance of the proposed solution is given. • Metrics and/or benchmarks are defined for assessment of the solution. • Evaluation results are presented • Any issues are discussed.	20
Conclusion	• Key lessons learned about the problem and solution are presented. • Future work / next steps are described	10
References	• References are complete • References are cited inline in IEEE style • Reference section is an IEEE style list of cited sources	10
Appendix: Personal Contribution and Lessons Learned (one per person working on project)	• Each team member must provide a 1-page summary of their key contributions to the project and their personal lessons learned from the experience. • For solo projects, simply discuss the lessons learned from the experience.	-10 point penalty if missing (for that person)
Writing Quality	• Professionally formatted • Professionally organized • Typo and grammatical error free	10