CS303 Project1 Report

Structure

A report should at least include the following sections. The order given here is suggested but not necessary. You can re-organize it as long as the expected information is presented.

1. Introduction

1. Give a general introduction to the problem studied in this project. For example, where does it originate from, how can it be characterized, and to what kind of real-world problems can it be applied?
2. State the purpose of this project/report.

2. Preliminary

Formally formulate the problem, and explain the terminology and notation you will use throughout this report.

- A *formulation* is an abstract but accurate description of the problem. It should disambiguate the potential confusion in natural languages.
- Example: The problem can be formulated as a Markov Decision Process, which is specified by a tuple $(\mathcal{A},\mathcal{S},\mathcal{T},r,\gamma)$, where $\mathcal{A}$ is the action space, ..., and the objective of an agent is to maximize $\sum_{t=1}^T r(s_t,a_t)$.

3. Methodology

1. General workflow.
    - Example: The proposed method is divided into steps 1, 2, and 3, each involving algorithms A, B, and C, respectively.
2. Detailed algorithm/model design.
    - Describe the **essential part** of your algorithm/model with pseudo-code/flow charts/diagrams.
    - **DO NOT paste (edited) Python code**.
    - If there is some complex data structure that is not intuitive to understand how it is implemented, give additional explanations.
3. Analysis. Discuss, for example, the optimality and complexity of your algorithm, and what is the deciding factor of its performance.

4. Experiments

1. Setup.
    - Give a short introduction to the dataset(s) you used (if any). Try to comment on the dataset's characteristics and relate them to your analysis. For example, if your dataset contains graphs of different sizes, make a table or histogram to show statistics on the edge and vertex numbers they have.
    - If you generated data by yourself, briefly talk about how.
    - If you find something from the Internet, indicate the source.
    - Describe the environment, e.g., software/hardware configuration, Python and NumPy versions.

2. Results.
    - State the performance measures of interest, e.g., running time, optimality.
    - Experimental results (using tables, figures, charts, etc.)
    - Try to find through the experiments:
        - the effect of different optional components of your algorithm (if any), e.g., Flip bit mutation, Inversion mutation.
        - the effect of hyperparameters (if any), e.g., the population size in genetic algorithms.
3. Analysis.
    - Comment on the experimental results. Are they good or bad? Do they meet your expectations/hypothesis?
    - Analyze the effect of different components and hyperparameters if you have corresponding experiments.
    - Try to relate the results to the **Methodology** part. For example, discuss the relationship between theoretical time complexity and the actual running time, give possible explanations for the discrepancy (if any).

5. Conclusion

Draw **informative** conclusions from what you have done and written.

Possible things you can write:

- Comments on the advantages/disadvantages of the algorithm you used.
- Does the experimental result match our expectations/analysis?
- The lesson you learned from this project. E.g., how to get a fast implementation in Python?
- Further thoughts on how it can be improved.
- ...

Tips

You can write the report in Word, Markdown, or LaTeX, but the submission must be in PDF format.

If you choose LaTeX, here are some suggestions:

- Use Overleaf or ShareLaTeX provided by SUSTech CRA to write your reports/articles in LaTeX.
- To get an IEEE article template:
    1. Visit the template selector page.
    2. Make the necessary selections and download the template as a `.zip` file.
    3. Log into Overleaf or ShareLaTeX, select `New Project` -> `Upload Project`, and upload the downloaded `.zip` file.
- For writing pseudocode in LaTeX, refer to Pseudo-code in LaTeX.