CS 634 Data Mining
Final Term Project

Yasser Abduallah
Department of Computer Science
New Jersey Institute of Technology

NJIT
New Jersey's Science & Technology University

THE EDGE IN KNOWLEDGE

Table of Contents

General Submission Rules

- Embed your last name and first name in your project file name. For example, if your name is John Smith, your file name should read: smith_john_finaltermproj.doc. Only doc or pdf file is accepted. No tar/zip/rar is allowed.

- Your project will automatically lose **10** points if the above submission rules are violated.

- Submit your project file in Canvas under Final Term Project Submission Site before the due time. The project file in Canvas is considered as the final version.

- No late project is accepted. A project is late if it is not submitted in Canvas before the due time. Zero points will be given to the late project.

- **NOTE: Pay attention to each option submission rules, there might be additional rules and requirements.**

Project Grading

- The grades will be posted on Canvas when they are completed.

- Note: There is a limit on the file size in Canvas and in NJIT’s email box. So, keep your project file small to avoid any problem that may occur when submitting the file in Canvas.

- The project file must contain the **all source code** and documentation including **screenshots**. The screenshots are used to demonstrate the running situation of your program, particularly how the program executes and produces output based on different input data and user-specified parameter values, if applicable.

- Github & Jupyter Book.
  - After you finish your code in development and testing and make sure it works, and prepared the report (meaning all heavy lifting job is done 😊), Create a Github repository in https://github.com/. Your account must be with your NJIT email not your personal email.

  - Load your project to the repository.
  - Create Jupyter book for your work to show the output, for more info visit https://jupyter.org/

  - Give me ya54@njit.edu access as a collaborator to your repository. (If we have a grader, you give him/her access too).
  - Add Github link to your repository to your report.
    NOTE: If you need help with Github and/or Jupyter book, let me know.

- Project milestone is a mid-way to show your progress.

  Submit a one-page project proposal that includes:

  1. Which option you selected with some details.
  2. What software you are planning to use.
  3. Hardware configuration.
  4. Framework, if applied.

- Copying and sharing code with peers is prohibited and will result in 0 point for all parties that are involved.

Final Term Project

- This is a single person project. *Do not share or copy code from your peers.*
- There are several options available for this project. You need to select only one of them to implement.
- Make sure to follow the submission rules when submitting your project.
- The Appendix will include a list of helping documents, sites, tools, and resources, for each option, will be available to help you implement your project. Make sure to read and use these documentation and resources.

Option 1: Supervised Data Mining (Classification)

- Implement 3 different classification algorithms in Python. One of them is Random Forest and the other two are from the list of algorithms in "Appendix → Option 1" on 1 dataset of your choice (each of the three algorithms must run on the same dataset).

  You may also make it fun to try to solve one of an existing problems:

  ie: Quora Insincere Questions Classification, Predicting Diabetes from Medical Records

  NOTE: This is not from scratch implementation, just use the existing libraries.
- Sources of data are listed in the Appendix "Option 1: Sources of Data", or use your own.
- Your final term project documentation must clearly indicate the algorithms and dataset you used in the project.
- In addition to the general submission rules and grading, include the websites where the software and complete dataset can be downloaded.
- You must present experimental results that show the comparison of classification accuracies between the classification algorithms used in the project.
- In evaluating classification accuracy, Student must use the 10-fold cross validation method (if your algorithms predict labels) or present ROC and AUC (if your algorithms produce continuous values). You must show the statistics as discussed in the "Evaluating Classifiers" module, that include all parameters that were introduced: TP, TF, FP, FN, TSS, HSS, etc.. for each run of the 10-folds and also for overall as an average of all 10-folds execution.
- Provide your final result in tabular format listing all details for easier visualization. Your Juypter Notebook should also show the final result table.

Option 2: Deep Learning

- This option is to implement 1 classification (or any deep learning) algorithm of your choice on 1 dataset of your choice using the deep learning software library TensorFlow or PyTorch.
- Your term project documentation must indicate clearly the algorithm and dataset you used in the project.
- In addition to the general submission rules and grading, include the source code for the deep learning implementation.
- Indicate any third-party library you used within your implementation.
- **Note: Google has free deep learning cloud for research https://colab.research.google.com/notebooks/intro.ipynb you can use it if you want, or you can also leverage NJIT data science server lochness/kongt**
- **NOTE: Please, you need to discuss with me first if your choice is accepted or not**

Option 3: Frequent Pattern Tree in Spark

- This option is to implement the Frequent Pattern Tree association algorithm using Apache Spark.
- You may implement using Java or Python.
- Your term project documentation must indicate clearly the algorithm and dataset you used in the project.
- In addition to the general submission rules and grading, include the source code for the deep learning implementation.
- Indicate any third-party library you used within your implementation.
- Include screenshot of the Apache Spark Environment you use, configuration, number of nodes (master, slaves), number of JVM if running in single machine such as your laptop, etc.. the more information the better.
- Screen shot of running Spark commands.
- Configure Spark to write the result to file and include the result to your report.

Infrastructure and Framework

- You may use NJIT Apache Spark (if this not available use the following options and discuss it with me how)
- You may also use Apache on your own machines, but make sure to run it in cluster mode not standalone mode.

NOTE 1: This is should be your own implementation of FP-tree in Spark, if you use the existing Spark ML library for FP-tree, some points will be subtracted from the grade, but talk to me first to discuss if points should be subtracted or not.

NOTE 2: You should start working on this option as early as possible, at least start working on Apache Spark Environment set up to be ready for FP-Tree when the module becomes available.

Appendix

Option 1: General Sources of Algorithms/Software

- https://scikit-learn.org/stable/

Option 1: Algorithms:

- Select two classification algorithms from the following:

I. Algorithm (Support Vector Machines)

II. Algorithm (Decision Trees)

III. Algorithm (KNN, K-Nearest Neighbor)

IV. Algorithm (Bayesian Networks)

V. Algorithm (Naïve Bayes)

VI. Algorithm (LSTM – Deep Learning)

Option 1: Sources of Data

1. http://aws.amazon.com/datasets
2. https://archive.ics.uci.edu/ml/index.php
3. And more you can find or your own…

Option 2: General Sources of Algorithms/Software

- https://www.tensorflow.org/get_started/
- http://deeplearning.net/
- https://pytorch.org/

  You will find a lot of examples on deep learning implementation, please don't copy and paste, understand and implement, or else you will lose the grade for the project.

Option 3: General Sources of Algorithms/Software

- https://spark.apache.org/
- https://scikit-learn.org/stable/