Rochester Institute of Technology

CSEC 520/620: Cyber Analytics & Machine Learning

Project: Data Process & Application

Due Date - December 12, 2021 11:59 PM

Purpose

The purpose of this project is that you and your team will apply what you have learned in this course to process a raw set of data into a form usable by ML algorithms and evaluate your processed data in some security application (e.g. classification tasks, unsupervised anomaly detection, etc). This project is open-ended; you will select the dataset and ML techniques to use on your own.

Description

Tasks you will need to accomplish in this project:

1. Identify a suitable repository of **raw data** to use for the project.
   a. To ensure that your selected raw data is suitable, you must check it with the course TA for guidance. This is due by **November 23, 2021, 11:59 PM**. Failure to submit this on time will result in a project grade of zero.
   b. This data should be 'raw.' What we mean by that is the current format of the data should be unsuitable for machine learning techniques as is. The data should require several steps to process into a usable format.
   c. Some example datasets:

- The original format of the IoT dataset that was prepared for you in Assignment 4 is a good example of raw data. This dataset is publicly available in the form of several large PCAP capture files, where each capture file contains all the network traffic present on the network for a single day. To process the raw data into a usable format for classification, individual network flows needed to be extracted from each large capture.
    - The raw traffic samples are accessible here;
    - The processed results are here;
    - The scripts used to perform the processing here.

  This dataset is for example only, **you may not use it for your project**.

- The network traffic datasets used to evaluate website fingerprinting attacks are also fine examples of unprocessed data. These datasets are usually prepared as plaintext files that contain a series of packet descriptors for each network packet produced when visiting a website. To use this data in ML techniques, this data must be processed into a set of features (e.g. total traffic length, average packet size, etc).
    - The knndata.zip file available here is an early dataset often used in this domain.
- The Sherlock dataset contains time series data for sensors on different mobile devices

provided by volunteers. To use this dataset effectively, you would need to first determine what type of security application you want to build using the dataset (e.g. malware detection). Next, you would need to determine a scheme to process the data such that labels and feature information are available to use with an ML classifier.

- There are many sources for raw data. Here are is a couple of user-maintained lists of datasets that may be of interest:
    - https://www.unb.ca/cic/datasets/index.html
    - https://github.com/shramos/Awesome-Cybersecurity-Datasets
    - https://vizsec.org/data/

2. Design data processing techniques to turn your dataset into a usable form.
    a. You should consider what type of application and ML techniques you intend to use when you evaluate your dataset.
    b. Take note of what techniques work well and what techniques fail. Both will be interesting information to include and discuss in your report and presentation.
    c. You should perform some data analysis on your dataset. This could include techniques such as generating statistics for your data and clustering.
3. Train and evaluate ML models on your processed dataset.
    a. Identify what metrics you will use to evaluate the performance of your models on your dataset. The metrics you use may be dependent on your security application. You should have reasons as to why you've selected your metrics, don't just report every possible metric.
    b. If possible, you should try to evaluate your dataset against several appropriate ML models.
    c. You may use third-party libraries such as scikit-learn to build your ML models. You may use models you've prepared in previous assignments, but you are also encouraged to try models that haven't been explored previously.

- Unlike the previous assignments, this assignment is not guided. You must decide for yourself what dataset and techniques to use to complete this assignment.

- You should identify and obtain the data you want to use for this project as soon as possible, as some datasets may require you to perform additional steps to request and download the data.

Deliverables

1. All source code of your implementations.
    - Document your code! Your code documentation should demonstrate you understand what your code is doing (and that you did not just copy-and-paste from an external source).
2. A readme file that should contain:.
    - A clear description of the directions to set up and run your code.
    - If your code requires external libraries, or if not written in Python, provide the additional references/direction.

   If your instructions are not clear, your work will ***NOT*** *be graded!*
3. **A** formal report describing your data processing, experiments, and results of your evaluations.

4. Prepare a short presentation (~20 minutes for slides and Q&A) to be given during final exam week that details your experience and results for the project. More information will be forthcoming about the presentation requirements and grading.

Expected Report Elements:

1. Submit a formal academic report of your project. If you submit your code's documentation or a README file or a notebook (ipynb as html/pdf) file as your report, then it won't be graded (meaning, the *project grade will be zero*).
2. The report should provide the details of the following:
    a. Some data analysis on your dataset (e.g., generating statistics for your data and clustering).
    b. Data processing techniques that you used to turn your raw dataset into a usable form.
    c. The ML techniques you intend to use when you evaluate your dataset.
    d. The metrics you use may be dependent on your security application. You should have reasons as to why you've selected your metrics
    e. The performances of the different ML algorithms and the analysis of those scores.
3. Include any citations in an appropriate and consistent format.
4. **Not Expected:** Background on the ML methods, the paper, or the dataset. Especially if it leads you to copy other material (see "A Note on Plagiarism" below!).

Grading Rubric

Criteria	1 Poor	2 Basic	3 Proficient	4 Distinguished
Dataset Selection & Processing (30%)	Choice of data and/or data processing is inappropriate or nonsensical.	Acceptable choice of data, however data processing technique may be too simple or lacking in depth and effort.	An interesting dataset was selected and the data processing techniques used are solid. However some details may be unclear or may be lacking appropriate data analysis.	Excellent choice of data and processing techniques. The dataset and processing techniques are very well described and investigated.
Experiments & Results (25%)	Minimal results are reported. Experiment details are undescribed.	Results are incomplete/lacking in detail, or are not presented in a way where its meaning is clear.	A reasonable set of results showing expected performance. Some presentation issues such as confusing graphs/tables or unnecessary detail.	The models can achieve good performance and all experiments are presented clearly.
Analysis & Discussion (25%)	No or nonsensical analysis.	Analysis is inaccurate or hard to understand. Analysis is lacking in depth and detail.	Analysis is sensible and has depth. May be missing appropriate references or is weak in some areas.	Clear and accurate analysis that is backed up with appropriate references.
Writing Quality (10%)	Very poorly written and hard to follow.	Major points are visible, but writing may include many errors and/or lack focus and is disorganized.	Writing is clear enough to be understood, but some points may lack focus. Relatively few writing errors.	The paper is clear and well organized. Writing is smooth and polished with very few errors.
Code Documentation (10%)	Inadequate documentation provided both in the code and with the code.	Acceptable documentation for running the code, but lacking in-code documentation and helpful descriptions to demonstrate understanding.	Code includes some documentation (e.g. function docstrings, inline comments), however quality may be weak or unclear.	High quality documentation is provided. The purpose and function of all segments of code can easily be understood.

A: 3.25 average or higher

B: 2.5 - 3.25 average.

C: 1.75 - 2.5 average.

A Note on Plagiarism

When writing, you must include in-line citations whenever you are reporting on information that is not “general knowledge,” i.e. anything you learned for this project and didn’t know in advance. This is **NOT** just for quoted information. Failure to do this is plagiarism.

This article on plagiarism is good and covers the line between common knowledge and other material: https://writingcenter.unc.edu/tips-and-tools/plagiarism/. Also: https://www.plagiarism.org/ has a ton of additional information.

I have had students fail to follow these guidelines and get caught nearly every time I’ve taught my research seminar. These students get put on probation and have even been suspended from the university for this serious academic violation. *Please do not be the next!*