MACQUARIE University

Faculty of Science & Engineering

COMP8325 Applications of Artificial Intelligence for Cyber Security

Assignment 1 Part 2 Description
Total Marks: 12
Weighting: 12%

Deadline: **Monday, May 29, 11.55 pm.**

LEARNING OUTCOME

This assignment deals with the recovery of digital evidence. On successful completion, you will be able to

- Engage with the material learned in COMP8325;
- Explain the basic concepts and the limitations of Artificial Intelligence;
- Detect intrusion in networks and systems by applying tools and techniques revealing abnormal patterns in datasets; and
- Analyse the trends of applications of Artificial Intelligence in cyber security.
- Learn to use the Python packages for machine learning, e.g., `scikit-learn`, `numpy`, and `pandas`.

TASK 1: Merits of Entropy in Attack Detection/Diagnostics (marks 4)

Please answer the following questions based on a server-log dataset that is available on Google Drive at this link: https://drive.google.com/file/d/1JLMpm6aQ5FJWtBW0VhFBP-BFQ9lXCS4W/view?usp=sharingF. The dataset contains information about two attacks that occurred sometime between 8:00 am and noon on a single day:

- Identify the precise date and time of the attacks, as indicated in the columns of the dataset. Describe the attack methodology used by the attackers. **(marks 2)**
- There is a significant body of literature[1][2] that discuss the use of entropy to detect network attacks. Typically, approximation schemes are utilized to make this process more effective. It is not necessary to implement these approximation techniques, but an analysis should be conducted to determine the usefulness of entropy and the combinations of factors that should be tried, such as source IP,

[1] Lall, et all 2013. Data Streaming Algorithms for Estimating Entropy of Network Traffic,
[2] Clifford, Cosma, 2013. A simple sketching algorithm for entropy estimation over streaming data

destination IP, source port, and destination port. During the two attacks mentioned in the dataset, were there any anomalies revealed by any of these combinations? **(marks 2)**

TASK 2: Web Tracking (marks 8)

When a user accesses a webpage in their web browser, the webpage typically contains multiple web-components, such as images, JavaScript codes, Flash content, CSS, etc. These components are often downloaded through additional HTTP(S) connections from either the first-party domain (the website the user is visiting) or from third-party domains. This article will focus specifically on JavaScript codes, which are commonly used by ad networks, content distribution networks (CDNs), tracking services, analytics platforms, and online social networks, including Facebook's use of them for plugins.

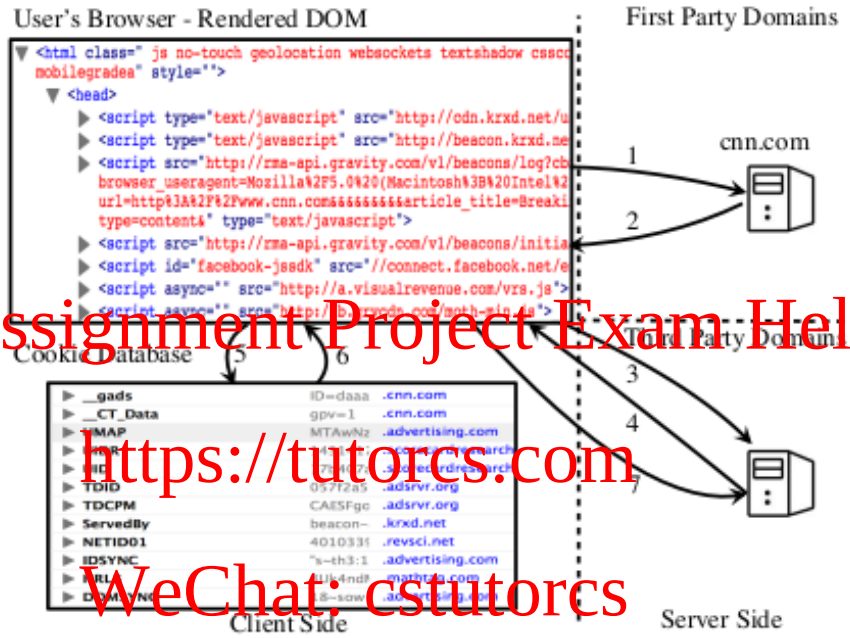

FIGURE 1. Overview of a webpage rendering process and web tracking. Websites (in this case `cnn.com`) use third-party domains for content provisions and analytics services.

The scenario of web tracking via JavaScript codes is depicted in Figure 1. When a user accesses a webpage from a first-party domain (steps 1 2), the web browser interprets the HTML tags and executes any JavaScript programs within the HTML script tags. These programs may trigger the browser to send additional requests to retrieve content from third-party domains (step 3). Depending on their intended functionality, JavaScript programs can be considered either useful (functional), such as fetching content from a CDN, or used for tracking purposes. In the latter case, once the webpage has fully loaded (step 4), the JavaScript codes track the user's activities on the webpage, read from or write to the cookie database (steps 5 6), and potentially reconstruct user identifiers. Tracking JavaScript programs may also be employed to "fingerprint" the user's browser and system, and transfer sensitive information to third-party domains (step 7).

Suppose you are tasked with developing a machine learning model based on a single class (e.g., One-Class SVM (OCSVM) or Positive Unlabeled (PU) Learning, see ref[3]) to distinguish between functional and tracking JavaScript codes. You will be provided with a labeled dataset that contains functional and tracking JavaScript codes, which can be found on COMP8325's iLearn page. You may use the code provided on iLearn to perform the following tasks.

[3]Ikram et al., 'Towards Seamless Tracking-Free Web: Improved Detection of Trackers via One-class Learning', https://imikr4m.github.io/files/pets_tracking.pdf

- Use Term Frequency - Inverse Document Frequency (TF-IDF) to extract features from functional and tracking JavaScript codes. **(marks 2)**
- Develop either One-Class SVM or PU Learning, and a baseline SVM for comparison, to classify the JavaScript codes. **(marks 3)**
- Design and conduct experiments to validate and test the efficacy of your developed model **((marks 3)**:
  - To report any over- or under-fitting of the models, you may use 60% of the data for testing, 20% for validation, and 20% for the testing.
  - Report and discuss the parameters of OCSVM or PU Learning model which give your improved results.

SUBMISSION

You will submit a zip/tar file that includes:

- An individual report discussing the process and results of applying the machine learning models for the specified data analysis tasks. You need to briefly depict the machine learning models you use to a data set. Then, you are required to describe and justify the process of handling data pre-processing, feature selection/extraction, and parameter tuning. Data analysis results with tables or figures should be reported, together with your interpretation and critical thinking.
- The source code (in Python) can be executed with ease. No GUI is required, but clear README or other documentation should be given. This means that we can repeat your data analysis process and results with little effort. Another reason is that we will use the reserved testing datasets to justify your trained model again to see whether the testing errors comply with your results derived from the data sets released to you. You could also directly submit a GitHub link if you have your code there. In this case, just put the link in a noticeable place in the report, e.g., a separate/dedicated section. The code repository must be private and you need to invite me as a collaborator so that we can access the code. My GitHub account is: xinbx4u.

EXPECTATION AND TIMELINE

- Students should submit a single Word or PDF file.
- The assignment is due **Monday, 29 May, 11:55 pm**.
- Late submissions will incur the following penalties in accordance with the assignment policy detailed in COMP8325's unit guide.
- If you have a legitimate reason for submitting late, discuss this with the convenor well in advance of the assignment due date.

MARKING RUBRIC

Marks will be available in iLearn by one week after the submission due date.

- All the required data analysis tasks have been reasonably accomplished.
- The organization, presentation, and readability of the report
- Appropriate justification of which you have chosen and what you have done in the data analysis process, as well as critical thinking and understanding of the related aspects of the machine learning methods
- The quality of source code, especially the ease of using the code to perform prediction/testing on the reserved data.
- The prediction results in the testing stage (based on the reserved data sets).