Simple Model Assemblages for Website Identification

Charles Hu czhu06@wm.edu William & Mary Williamsburg, Virginia, USA

ABSTRACT

A clear and well-documented LTEX document is presented as an article formatted for publication by ACM in a conference proceedings or journal publication. Based on the "acmart" document class, this article presents and explains many of the common variations, as well as many of the formatting elements an author may use in the preparation of the documentation of their work. A cite so this compiles.

KEYWORDS

Network traffic, Network traffic classification, Machine learning, Ensemble models

1 INTRODUCTION

2 PROPOSED METHOD

The following section details the design and rationale behind the methodology used for evaluating and classifying the website of origin for given samples of web traffic using machine learning models.

2.1 Data Preparation

In order to ensure that our developed models are robust and effective at their classification task, we employ a measured and systemized approach towards data collection and preprocessing to ensure that possibility of gathering ideosyncratic or erroneous data is minimized as much as possible.

The primary objective during the design of the data preparation phase was to ensure that the generated data set allows us to construct models that are effectively generalizable and not too overfit. A key principle that allows us to achieve this is confirming that the gathered sample is as representative as possible of the general population; however, this is difficult to verify empirically given the complex nature of the overall population. As a result, a preemptive approach was taken which resulted in the design of a two-stage data preparation phase comprised of a precautious data collection stage and a mitigative data preprocessing stage.

2.1.1 Data Collection. Data collection is performed through the monitoring of artificial website activity aimed at emulating real user interactions common to the sampled websites. Website traffic between the user and the server hosting the

website is monitored and tracked using Wireshark, an opensource software used for network traffic capture and analysis [1]. Wireshark will specifically target the transport layer of network communication and intercept ongoing TCP streams between the user and the site host. Activity on a website will occur through the controlled usage of a website's typical functionalities. For instance, on a streaming site, the data collector will utilize the site's recommendation algorithm to view a certain number of videos before halting activity. The goal during the activity substage is to interact with the website in a naturalistic manner akin to any typical user but avoid operations which either go beyond the scope of the target website (e.g., entering another website through an embedded link) or are unexpected of a user (e.g., manually performing HTTP operations with the website or abruptly closing the site during an interaction). These precautions should help minimize the amount of collected TCP streams that contain information that are erroneous or overly noisy. Once activity on a website has been concluded, Wireshark will be used to display the TCP streams generated during activity with the website (as seen in Fig. 1) and will export the TCP streams to an external CSV file for each targeted website.

- 2.1.2 Data Preprocessing. Data preprocessing involves pruning the resulting data set from the data collection stage to remove features that are either explicitly detrimental or extraneous. The removal of these features help improve the generalizability of our developed models by decreasing the overall model complexity and thus the variance inherent to it [2]; thus, if any idiosyncrasies or noise make it through our first stage of data collection, we can still reduce their influence on the overall model performance by lowering the capacity of the model to learn bad data. To ensure that we do not develop models that have too low complexity due to a lack of learned features (and therefore trending towards too much bias in the bias-variance tradeoff), only features that are explicitly detrimental or extraneous will be removed. Removed features are as follows:
 - Address A, Port A, Address B, Port B: These feature are explicit identifiers for the hosts in the TCP stream. Their inclusion would make the classification task redundant and would result in a model that

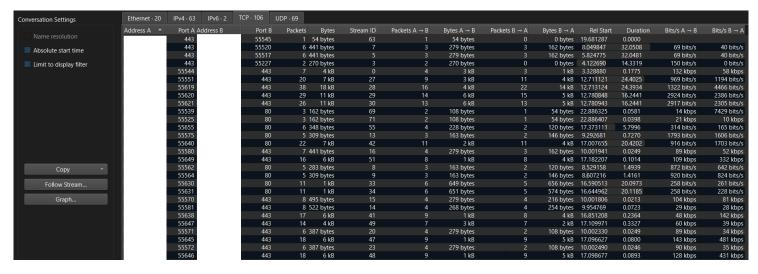


Figure 1: A Wireshark window displaying tracked TCP streams and each stream's corresponding attributes. Note that IP addresses have been censored.

places high importance on observing the IP and port values over other potentially useful features.

- **Stream ID**: This feature tracks the unique internal tracking ID given by Wireshark to any given TCP stream for later reference. This is Wireshark specific and not pertinent to the TCP stream itself.
- Total Packets, Percent Filtered: These features refer to the fitler display feature in Wireshark which filters every available packet to search for some filter specification (in this case, the target website). These are Wireshark specific and not pertinent to the TCP stream itself.
- **Rel Start**: This feature indicates when the TCP stream began relative to the start of the Wireshark network capture session. This is Wireshark specific and not pertinent to the TCP stream itself.

Retained features from the raw data set are as follows:

- Packets: This feature tracks the total packets transferred. Note that since this is a TCP stream, this feature instead referes to a TCP segment.
- Bytes:
- Packets $A \rightarrow B$, Bytes $A \rightarrow B$:
- Packets $B \rightarrow A$, Bytes $B \rightarrow A$:
- Duration:
- Bits/s A \rightarrow B, Bits/s B \rightarrow A:

2.2 Data Specifications

2.2.1 Data Analysis.

2.3 Model Development

2.4 Model Specifications

- 2.4.1 Simple Models.
- 2.4.2 Ensemble Models.

3 EVALUATION

3.1 Evaluation Metric

- 3.2 Results
- 3.2.1 Baseline.
- 3.2.2 Simple Model Results.
- 3.2.3 Ensemble Model Results.
- 3.2.4 Model Results Overall.

4 DISCUSSION & FUTURE WORK

4.1 Result Interpretations

- 4.2 Future Work
- 5 CONCLUSION

REFERENCES

- [1] Wireshark Foundation. 2024. Wireshark. https://www.wireshark.org/
- [2] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze.
 2008. "Introduction to Information Retrieval". Cambridge University Press.