Human Network Traffic Labeling System and a Comparison with Deep Packet Inspection

Michael P. McGarry, Herman F. Ramey, Luis Vergara-Rodriguez

***Abstract*—Network traffic classification is a core network intelligence to empower efficient network management (e.g., ensuring QoS requirements). Accurate training data for machine learning is a necessity to achieve accurate network traffic classification. Deep packet inspection to generate training data leads to vagueness that significantly compromises accuracy. Human-supplied ground truth is the solution to this problem. We present our human network application labeling system that contributes a new level of distinction between the network traffic that should be labeled from the network traffic that should not be labeled. This distinction improves the label accuracy of the training data set produced from the human labeled data and will subsequently improve the performance of supervised machine learning classifiers used for network traffic classification. This system also allows for the human network user to label traffic, with little effort, in a manner consistent with normal network usage, i.e., no need for a contrived experiment. Lastly, we use human supplied ground truth network application labels to analyze the performance of deep packet inspection techniques, specifically the nDPI library.**

***Index Terms*—network traffic classification, network traffic identification, network intelligence, supervised learning**

I. Introduction

Network traffic classification (NTC), also known as network application classification, can have broad impact across intelligent network management. NTC can be utilized to provide application-selective traffic management, aid in network security, add an additional feature to network traffic for various machine learning tasks, among other uses. As a specific example, NTC can be used to block unwanted network applications (e.g., remote login) or deliver personalized Quality of Service (QoS) features by prioritizing certain network applications over others [1].

Port number conventions can be used to classify network traffic and have been shown to possess strong accuracy in conditions where those conventions are followed [2]. The IANA website has a list of those port number conventions, see [3]. Deep packet inspection (DPI) uses payload signatures, and other data to classify network traffic [4] and has been shown to perform well when network traffic is not encrypted. Open source libraries such as OpenDPI [5] and nDPI [6] provide these functionalities. DPI, given its nature of inspecting the contents of packets, can raise significant privacy concerns [7], [1]. Supervised machine learning classification algorithms can perform well under a wider set of conditions when discretized network flow features such as the first few packet sizes are used [2]. However, these classification algorithms require training datasets to train the classifier or classifiers to later perform the classification on unknown traffic. Note: network traffic is represented as network flow data such as NetFlow v9 [8], [9].

Many authors highlight that building a training dataset that contains labels which are both accurate and representative of real network environments is crucial to yielding satisfactory results [10], [11]. However, acquiring enough labeled training data to build a high-performing classifier has proven to have challenges [12]. This may involve tasking a subject matter expert to manually label all traffic in a controlled network environment [13], which is potentially very time consuming when trying to build large enough datasets for classification purposes. Additionally, the inherent tendency for humans to make errors compared to automated techniques may consequently jeopardize accuracy of the labeling process, ultimately degrading the overall performance of the classifier.

A human labeling system that minimizes the human effort to provide application labels to network traffic, as represented by network flow data, is an ideal method for generating a training dataset to classify network traffic. This is the objective of our work but first we will review the existing literature on human application-labeling of network traffic.

A. Related Work

WebClass. Ringberg, Soule and Rexford [14] devised a technique to provide human generated traffic labels. This technique, known as WebClass, is a web-based software system that enables users to examine and label potential anomalies on time-series of traffic measurements through a graphical user interface. Their research harnessed a community of experts for annotating and continually monitoring labels, ensuring they remain current and highly precise. WebClass provides a platform for researchers to collaborate on the network traffic labeling process, aiming to enhance both its simplicity and reliability. By enabling users to review others' labels, the community could collectively identify and agree upon genuine positive anomalies. Focusing on labeling of anomalies, this system does not address the problem of generating training datasets for NTC.

UTMobileNetTraffic2021. In 2021, Heng, Chandrasekhar, and Andrews, with the help of six undergraduate students from UT Austin [15], created an automated platform designed to produce and label data traffic from numerous popular mobile apps within a controlled setting. Their system configures mobile devices to run only one application at a time and they subsequently collect all traffic to/from the device and assign the label provided by the human user, such as scrolling a newsfeed on Facebook or watching a video on Netflix, enabling

the data to be tagged with both the application name and the corresponding activity. The assumption that all traffic to/from the mobile device is associated with the labeled activity is quite weak; despite efforts to limit all other activity on the device it is very likely that some background processes are communicating with remote systems. As a result, it seems very likely that there will be traffic that will be mislabeled and compromise the accuracy of supervised machine learning trained on that data.

Automatic vs. Human-Guided Labeling. Jorge Luis Guerra, Carlos Catania, and Eduardo Veas conducted a study on the challenges presented in the labeling process, identifying two main approaches that exist today: automatic and human-guided. Their recent survey [10] on current labeling methods highlights the advantages and disadvantages of each approach. They delve deeper into the specific implementations of both automatic and human-guided techniques, such as i) injection timing versus behavior profiles for *automatic labeling*, and ii) manual versus assisted for *human-guided labeling*. According to their findings, using an automated labeling method involves generating a dataset within a structured and predictable network environment. This helps identify abnormal activities amidst regular traffic, thereby removing the need for expert manual labeling. On the other hand, when using human-guided labeling methods, the network environment lacks control, placing the responsibility entirely on expert users. Guerra et al. go on to discuss a significant challenge that presents itself when implementing the human-guided labeling technique, which is the difficulty in labeling the volume of traffic needed for current network intrusion detection requirements.

B. Our Contribution

Our human application labeling system addresses the shortcomings of the existing literature by very significantly improving the distinction between the network traffic that should be labeled with the human supplied label and which traffic should not be. Further, our labeling system operates in a way to allow the human network user to label traffic during their normal every day use of the network rather than as part of a contrived experiment. Lastly, we contribute to the literature by providing an analysis of deep packet inspection, specifically that provided by the nDPI library. We use genuine ground truth labels provided by a human user to facilitate this analysis that reveals and/or confirms the shortcoming of deep-packet inspection beyond the privacy concerns it invokes.

C. Outline

In Section II we present the design of our human application labeling system that improves accuracy compared to the literature. In Section III we analyze the effectiveness of nDPI-based deep packet inspection using the human labeled dataset as ground truth. Finally, in Section IV we summarize our findings and outline avenues for further investigation.

II. System Design

Our human application labeling system prompts the user for an application label when they trigger activity in a web browser and subsequently labels the appropriate network flow data with that human-supplied label. Figure 1 is a screenshot of this system in action. The system consists of a Firefox web browser extension (written in Javascript) and a set of Python scripts that interact with the browser extension to label the appropriate network flow data generated using the Promiscuous Mode Accounting (`pmacct`) [16] software package. Figure 2 illustrates the design of the system.

A user of a web browser initiates network activity by either entering a URL in the address bar, selecting a URL from a bookmark, or clicking on a URL link on a web page. We will refer to these URLs as an *original URL*. Subsequently, the original URL activates additional URLs embedded in the content of the original URL, we will refer to these URLs as *requested URLs*. The original URL activates a stream of URLs that include the original and several subsequent requested URLs. The URLs in the stream are not only for content relevant to the activity the user is interested in, such as video streaming, but also for other purposes, such as providing information to a data broker. We will refer to the former as *intended traffic* and the latter as *unintended traffic*. We designed the system carefully to distinguish between the *intended traffic* that should receive the human label and the *unintended traffic* that should not. To make that distinction, we use `WHOIS` queries on every host name in the stream of URLs emanating from the original human-generated URL. If the organization of a URL matches that of the original we assume this is intended traffic and the flow identifying fields (including IP addresses and port numbers) are used by the Python scripts to find the relevant network flow data produced by `pmacct` and label it. If the organizations do not match, then the traffic is assumed to be unintended and therefore not labeled.

A. Obtaining Human Label

The Firefox (developer edition) web browser [17] extension is written in Javascript and uses the WebExtensions API [18]. It traps the event when a user enters a URL in the address bar, selects a URL from a bookmark, or clicks on a URL link on a web page; i.e., the *original URL*. On this human initiated URL event, the browser extension presents the user with a dialog box that prompts them to select an application category that matches their activity. This dialog box can be seen on the left-hand-side in Figure 1. Each URL in the full stream of URLs emanating from this *original URL* is logged, via a companion Python script, with the following information: timestamp, the requested URL, the original URL, the remote web server IP address and port number.

B. Collecting Flow Data

The `pmacct` software runs on the user's machine continuously collecting network flow data from the network interface. It consists of a flow exporter process and a flow collector process. We have configured the exporter to export NetFlow v9 data that includes: source and destination IP addresses, source and destination port numbers, flow start and end timestamps,

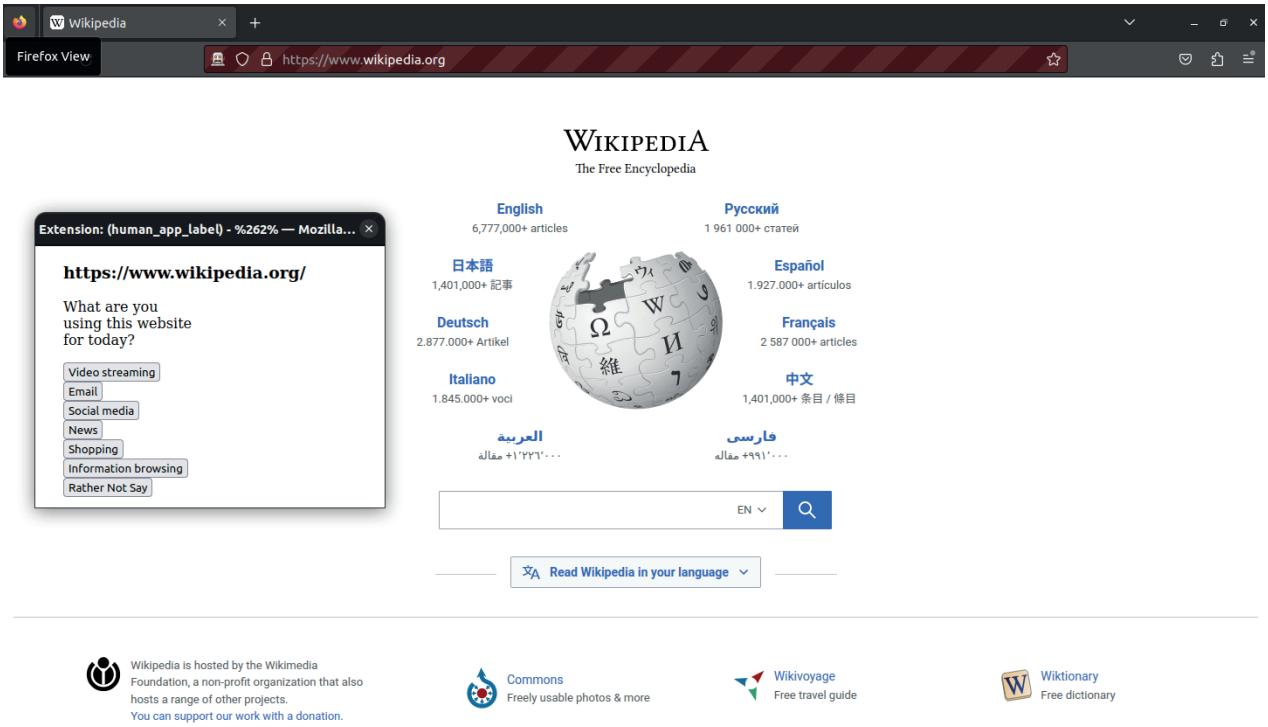


Fig. 1. Our Human Application Labeling system is a web browser extension that prompts the user for a label when they initiate a web request either through the address bar, selecting a bookmark, or by clicking on a link.

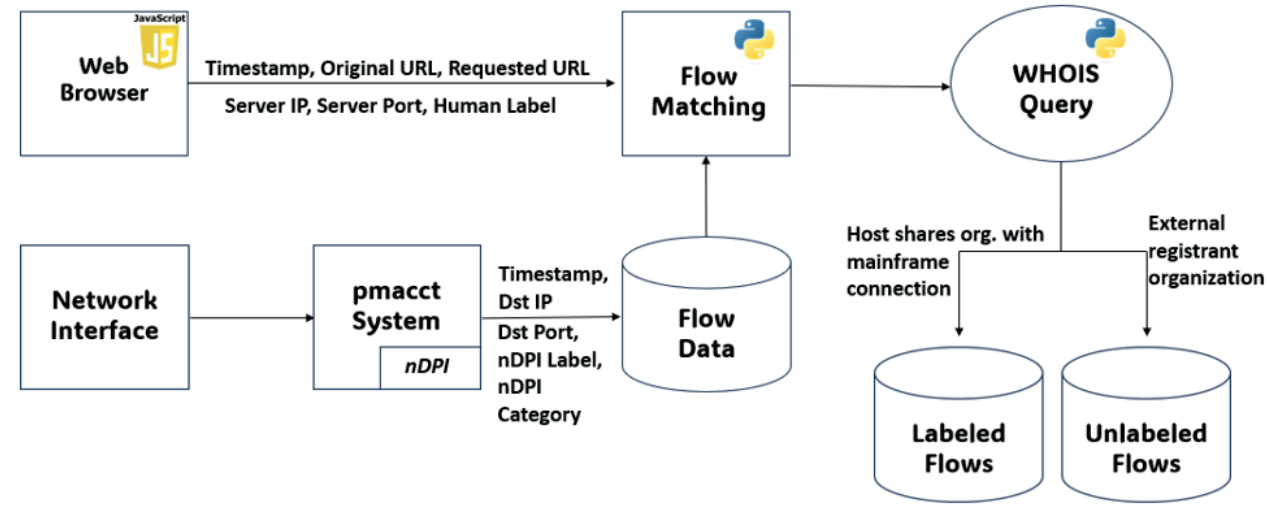


Fig. 2. The design of the Human App Labeling System. The system uses a Firefox browser extension written in Javascript to trap URL generation along with a human-generated application label. The stream of URLs with the human label are processed by a flow matching script written in Python that finds the corresponding pmacct-generated flows for each URL and attaches the human-generated label to that flow. WHOIS queries are used to filter the stream of URLs to ignore URLs that are not traffic intended by the user and should therefore not receive the human-generated application label.

as well as an application label. The application label is provided by deep-packet inspection using the nDPI library [6], [5]. This label is used by our experiments described in Section III to compare our human labels to deep packet inspection. The collector is configured to write all flow data to a comma-separated values (CSV) file.

C. Labeling the Correct Flows

We have written a set of Python scripts that process each generated URL logged by the browser extension and its companion Python script. We make the reasonable assumption that any communication with the organization of the original URL is *intended traffic* and any communication with a different organization is *unintended traffic*. To facilitate this distinction, for each URL entry, we verify if the URL is on a host associated with the original URL organization. This is more accurate than merely checking if the hostnames match; many organizations have several registered hostnames. As an example, both `netflix.com` and `nflximg.com` are registered to Netflix, Inc. We use a WHOIS query to obtain the registrant organization information. If the URL is determined to be intended traffic, its timestamp, remote web server IP address and port number are used to search for the relevant flows in the `pmacct` produced CSV file. Matching flows are labeled with the human label, adding to the label generated by the nDPI library.

III. Analysis of Deep Packet Inspection

We used our human application labeling system to label network traffic during normal network usage. As a result, we have network traffic, represented as NetFlow records, with two application labels: (i) the human ground truth label and (ii) the label provided by the nDPI library. We generated a total of 5,521 labeled network flows with applications in six application categories. We used the following six application categories that we felt meaningfully represented networked activities facilitated with a web browser:

- Video streaming
- Email
- Social media
- News
- Shopping
- Information browsing

A notable omission is the popular video conferencing application. This is omitted intentionally as that application is almost exclusively delivered outside of the web browser and therefore not covered by our human application labeling system.

Figure 3 shows a joint distribution of the human-generated ground truth labels and nDPI-generated labels, visualized as a heatmap. nDPI produces a two-level label: application and category. As an example, Netflix is the application and Video streaming is the category. The human-generated labels are at the category granularity by design. The first thing we notice from Figure 3 is that a vast majority of the flows are encrypted traffic and that nDPI fails to label those. Specifically, looking at the bottom row of that heatmap we can see that nDPI labels those flows as TLS while the human-generated labels reveal the application. This reveals the magnitude of a significant shortcoming of DPI: failing to classify encrypted traffic. A vast majority of the flows will, effectively, not receive an application label when using DPI.

Looking at the heatmap for *Social media* and *Video streaming*, the two far-right columns in Figure 3, we see that when not labeled TLS for encrypted, DPI does well at matching the human-generated labels. However, for Video streaming a vast majority of Video streaming flows are labeled as TLS by DPI. To get a more precise view of this we produced pie charts showing how flows (see Figure 4 plot a) and bytes (see plot b in that Figure) are distributed across the DPI-produced labels for Video streaming traffic. From those pie charts we can see that $\approx 67\%$ of flows and $\approx 81\%$ of bytes are labeled as TLS (i.e., encrypted) by DPI. Further, only $\approx 18\%$ of flows and $\approx 2\%$ of bytes are labeled with a DPI label in the category of Video streaming.

Moving our attention to the other application classes (Email, Information browsing, News, and Shopping) and looking back at Figure 3, we can see that the DPI labels are vague or are labeled as encrypted (i.e., TLS). The DPI labels, when not TLS, indicate the organization we are communicating with (e.g., Microsoft, Amazon, Cloudflare, Google) but do not distinguish the application class.

These results illustrate the two major shortcomings of DPI for NTC:

1) Does not label flows of encrypted traffic, which happens to be a vast majority of network flows
2) Produces vague labels for many application categories

However, it is worth noting that DPI has much better performance for Social media; with only about 20% of the flows labeled as TLS.

For a little more insight into these shortcomings of DPI, we produced heatmaps for more focused datasets. Figure 5 plot a shows the heatmap for a dataset whereby we streamed video from Netflix and Twitch and used Facebook. Figure 5 plot b shows the heatmap for a dataset whereby we streamed video from Hulu, browsed CNN's news website and used Instagram. We notice for Video streaming, that again a vast majority of those flows receive the TLS label from DPI. Upon close inspection of the flow data, we notice that the issue with labeling Video streaming is even worse than the heatmap suggests. Those few flows that are labeled as Video streaming by DPI, are very small flows in terms of packets or bytes. The large flows are labeled as TLS. It is likely those flows properly labeled as Video streaming are control messages not containing the streamed video content. If those labels were used to train a classifier, it would not correctly characterize the Video streaming traffic.

A. Discussion

The use of Deep Packet Inspection (DPI) to provide labels to create training data to be used for supervised learning seems to be limited to only Social media. DPI does not properly label

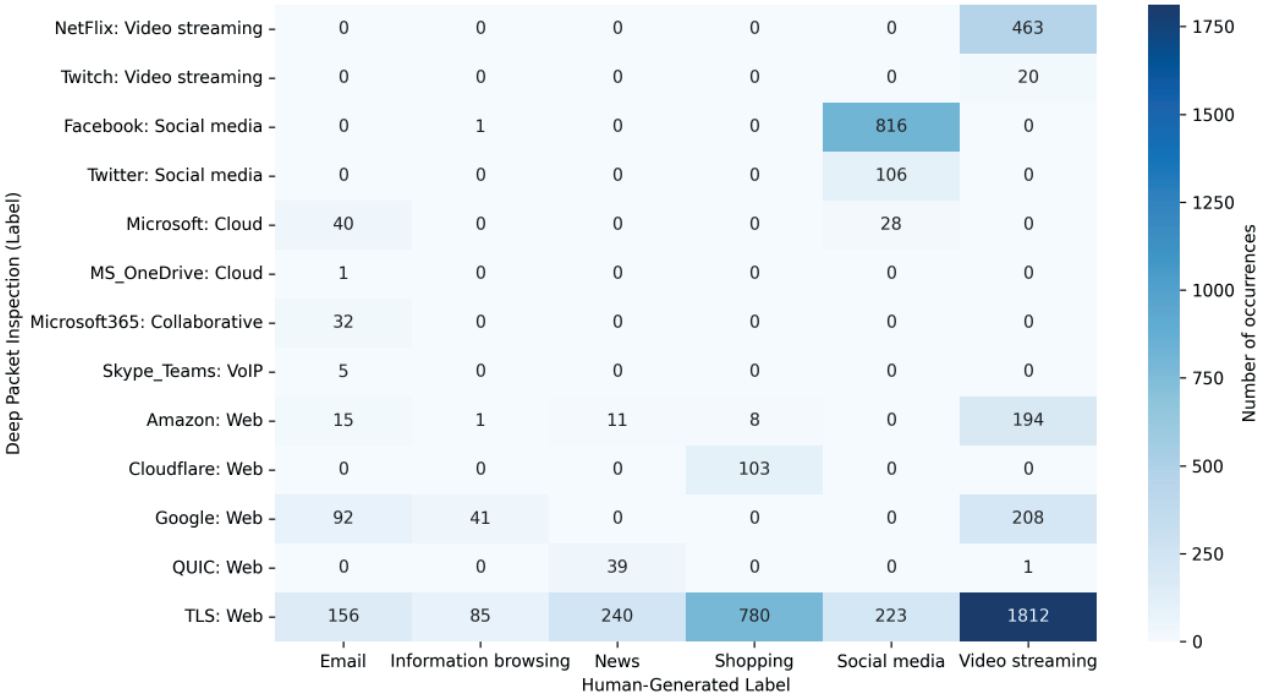
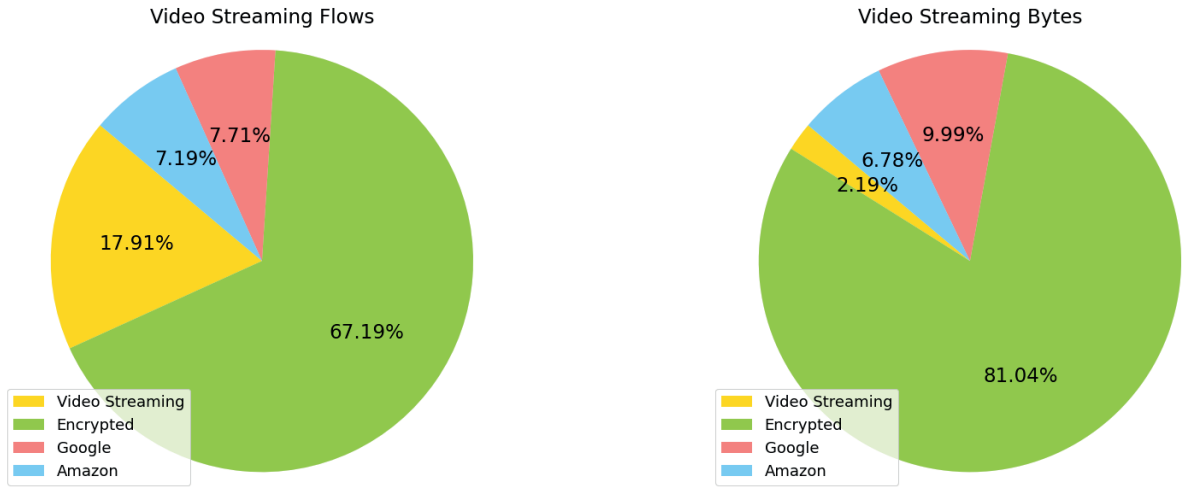


Fig. 3. Comparison of human-generated ground truth labels (x-axis) vs. corresponding DPI-generated labels (y-axis). This heatmap represents 5,521 flows.

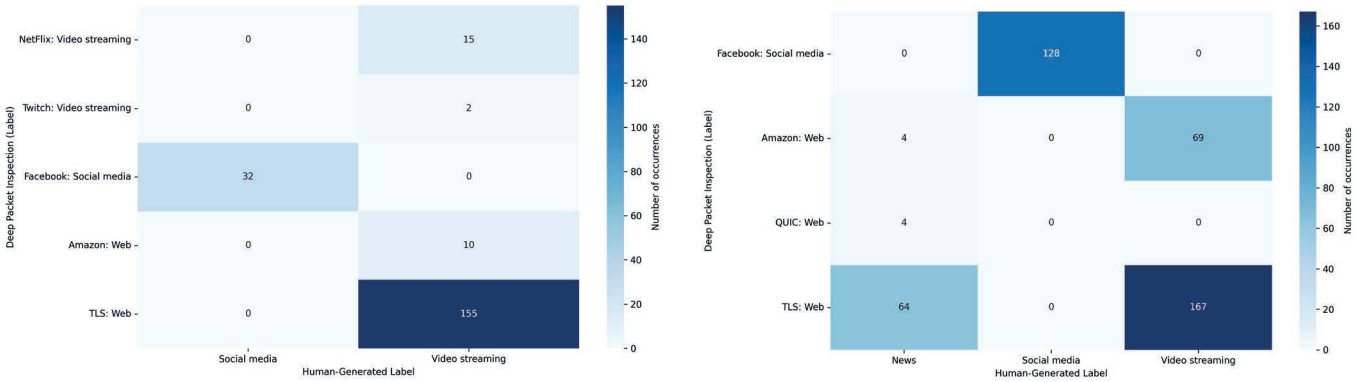


a) distribution of flows

b) distribution of bytes

Fig. 4. DPI traffic label distribution for Video streaming traffic. We can clearly see a vast majority of flows are labeled as encrypted (i.e., TLS). The majority is even larger when we consider the number of bytes.

the other categories of network traffic. The DPI provides a vague label either by only identifying the organization you are communicating with or specifying that it is encrypted traffic (i.e., the TLS label). These results suggest that training data for Network Traffic Classification (NTC) to be used to train supervised learning algorithms on flow features must be created using human-generated labels. Our software system that performs that function can meet this need in a way that requires only a minor effort from the human user. The lower effort will help incentivize the human user to provide the labeling but further incentives will also be required.



a) Streamed Netflix, streamed Twitch, and used Facebook.

b) Streamed Hulu, browsed CNN, and used Instagram.

Fig. 5. Comparison of human-generated ground truth labels (x-axis) vs. corresponding DPI-generated labels (y-axis) for two individual sessions with multiple activities.

IV. Conclusion

We developed and presented a human application labeling system that can be used to generate training datasets for supervised machine learning classifiers for network traffic classification (NTC). This system exceeds the capability of other similar systems reported in the literature in that it provides a strong distinction between the network traffic that should receive the human label and the network traffic that should not. Specifically, we identify the precise network flow data associated with the user activity and use WHOIS queries to distinguish between user intended network traffic (e.g., video streaming) and user unintended network traffic (e.g., providing information to data brokers).

We used the human generated ground truth network application labels to evaluate the performance of deep packet inspection (DPI) for application classification. Our results reveal how often network traffic is encrypted and therefore evades DPI and reveals instances where the DPI labels specifically from nDPI, are vague and therefore not as informative as the human supplied label from our system. The DPI labels only had some accuracy and precision for Social media traffic.

A few paths for future work:

1) utilize our human-labeled datasets to train supervised machine learning classifiers [19], [20] to classify traffic under a wide set of circumstances, compare performance to classifiers trained using nDPI-labeled datasets
2) explore methodologies that combine human generated ground truth with deep packet inspection
3) explore methods to incentivize human users [21], [22] to label the traffic

References

[1] J. Zhang, X. Chen, Y. Xiang, W. Zhou, and J. Wu, "Robust network traffic classification," *IEEE/ACM Transactions on Networking*, vol. 23, no. 4, pp. 1257–1270, Aug 2015.

[2] Y. Lim, H. Kim, J. Jeong, C. Kim, T. Kwon, and Y. Choi, "Internet traffic classification demystified: on the sources of the discriminative power," in *Proceedings of the 2010 ACM CoNEXT Conference*, Nov 2010.

[3] IANA, "IANA Service Name and Transport Protocol Port Number Registry." [Online]. Available: https://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml

[4] H. Kim, K. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, and K. Lee, "Internet traffic classification demystified: myths, caveats, and the best practices," in *Proceedings of the 2008 ACM CoNEXT Conference*, Dec 2008.

[5] T. Bhatia, "OpenDPI." [Online]. Available: https://github.com/thomasbhatia/OpenDPI

[6] ntop, "nDPI." [Online]. Available: https://github.com/ntop/nDPI

[7] C. Fuchs, "Societal and ideological impacts of deep packet inspection internet surveillance," *Information, Communication & Society*, vol. 16, no. 8, pp. 1328–1359, Oct 2013.

[8] R. Hofstede, P. Čeleda, B. Trammell, I. Drago, R. Sadre, A. Sperotto, and A. Pras, "Flow monitoring explained: From packet capture to data analysis with netflow and ipfix," *IEEE Communications Surveys and Tutorials*, vol. 16, no. 4, pp. 2037–2064, Oct 2014.

[9] G. Vormayr, J. Fabini, and T. Zseby, "Why are my flows different? a tutorial on flow exporters," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 2064–2103, Oct 2020.

[10] J. Guerra, C. Catania, and E. Veas, "Datasets are not enough: Challenges in labeling network traffic," *Computers and Security*, vol. 120, p. 102810, Sep 2022.

[11] F. Pacheco, E. Exposito, M. Gineste, C. Baudoin, and J. Aguilar, "Towards the deployment of machine learning solutions in network traffic classification: A systematic survey," *IEEE Communications Surveys and Tutorials*, vol. 21, no. 2, pp. 1988–2014, Apr 2019.

[12] M. Kim and I. Lee, "Human-guided auto-labeling for network traffic data: The gelm approach," *Neural Networks*, vol. 152, pp. 510–526, Aug 2022.

[13] M. Abbasi, A. Shahraki, and A. Taherkordi, "Deep learning for network traffic monitoring and analysis (ntma): A survey," *Computer Communications*, vol. 170, pp. 19–41, Mar 2021.

[14] H. Ringberg, A. Soule, and J. Rexford, "Webclass: adding rigor to manual labeling of traffic anomalies," *SIGCOMM Comput. Commun. Rev.*, vol. 38, pp. 35—-38, Jan 2008.

[15] Y. Heng, V. Chandrasekhar, and J. Andrews, "UTMobileNetTraffic2021: A labeled public network traffic dataset," *IEEE Networking Letters*, vol. 3, no. 3, pp. 156–160, Sep 2021.

[16] P. Lucente, "PMACCT: IP traffic accounting." [Online]. Available: https://github.com/pmacct/pmacct

[17] Mozilla, "Firefox Developer Edition." [Online]. Available: https://www.firefox.com/en-US/channel/desktop/developer/

[18] ——, "WebExtensions API." [Online]. Available: https://developer.mozilla.org/en-US/docs/Mozilla/Add-ons/WebExtensions

[19] J. Han, J. Pei, and H. Tong, *Data mining: concepts and techniques*. Morgan Kaufmann, 2022.

[20] Y. Bengio, I. Goodfellow, and A. Courville, *Deep learning*. MIT Press, 2017.

[21] A. Maslow, "A theory of human motivation," *Psychological review*, vol. 50, no. 4, p. 370, Jul.

[22] E. Fehr and A. Falk, "Psychological foundations of incentives," *European Economic Review*, vol. 46, no. 4, pp. 687–724, May 2002.