Interactive, Visual Uncertainty Quantification for Encrypted Network Traffic Situation Awareness

Harry X. Li, Allan B. Wollaber

MIT Lincoln Laboratory

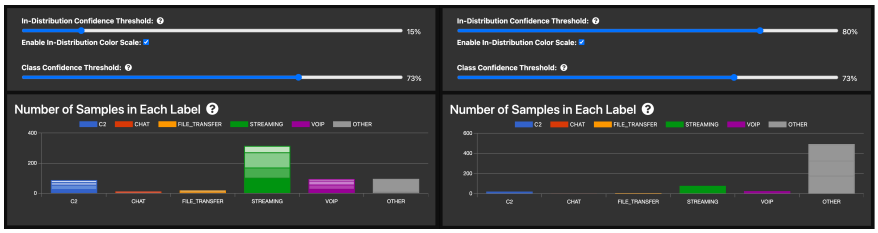

Figure 1: Dynamic uncertainty quantification example for epistemic uncertainty. On the left, the In-Distribution slider is set to a lower threshold, allowing the model to make predictions about the data, despite its uncertainty. The left bars have a light color, indicating a low In Distribution confidence. On the right, once the slider is set to a higher threshold, the dashboard moves the data into the OTHER category.

Abstract

As encrypted network traffic becomes increasingly prevalent, cyber network operators are operating in the dark with regard to the kinds of traffic flowing through their networks. Machine learning (ML) techniques have recently emerged that can rapidly learn and provide contextual labels to encrypted traffic. However, as ML-based applications reach the hands of operators, they do not always understand the limitations of the underlying models and their predictions, as ML models often struggle or fail to communicate the confidence of their predictions. Without this nuanced understanding, operators may blindly trust a model's predictions, unaware that the model is only marginally confident or has never seen the input data during training. We present a visualization dashboard for encrypted network traffic labels that combines confidence sliders and several visualizations to help contextualize the model's predictive confidence and the likelihood that the model has seen similar network traffic.

Keywords: Uncertainty quantification, machine learning

1 Introduction

Because traditional network traffic analysis techniques that rely on port mappings or packet payload inspection cannot provide situation awareness of the encrypted network traffic applications, machine learning (ML) models based on the observables of encrypted network traffic (e.g., timing and size statistics) have been developed to provide that context [1]. However, ML models suffer from two kinds of uncertainty, aleatoric and epistemic [2], that can adversely affect operator decision making. An example of aleatoric uncertainty in our model regards the confidence of assigning a label between two similar classes (say, sftp and scp) that the model was trained on. An example of epistemic uncertainty regards the confidence of the model assigning *any* label to an application it was not trained on. Network operators have situation awareness (SA) use cases for which uncertainty matters. For example, getting a rough idea of the proportions of network traffic for each IP address for quality of service would allow for some uncertainty, whereas scanning a network for any instances of a specific application would require higher confidence. We have created a prototype application that allows operators to quickly train their own models in order to gain SA of encrypted network traffic and here present how the prototype manages uncertainty via dynamic visualization. To our knowledge, this is a novel contribution to cybersecurity dashboards, as existing approaches are mostly concerned with 1- or 2-D data (error bars, image processing, geospatial, etc.) [3].

harry.li@ll.mit.edu
allan.wollaber@ll.mit.edu

2 Methods

Throughout, we use the (arbitrary) labels: C2 (Command and Control), CHAT, FILE_TRANSFER, STREAMING, and VOIP (Voice over Internet Protocol). To train a new model, users can load packet capture (PCAP) files into the pipeline and assign each file a label (eg., file1 represents CHAT, file2 is FILE_TRANSFER, etc). The new model is then trained on the labelled data, which can be used to make inferences on unseen PCAP data. For every data sample, the model assigns calibrated "Class Confidence" scores for each label (encoding aleatoric uncertainty) and an "In Distribution" score (encoding epistemic uncertainty) between 0% and 100%. The Class Confidence scores are the calibrated probabilities that the data sample belonged to each of the output labels, for example: CHAT: 25%,VOIP: 75%, etc. The In Distribution score encodes epistemic uncertainty [2], with 0% meaning that the ML model should not be trusted and 100% indicating that the new data is very familiar to the model.

To help operators explore and understand the limitations of their trained models, we created a visualization dashboard that has multiple ways for operators to view and filter the predicted labels of a network traffic dataset. Users first upload PCAP network data to the model, which classifies each data sample along with the In Distribution and Class Confidence scores. This labelled data is then returned to the dashboard to be visualized. In all of our visualizations, each predicted label is assigned a color (ie. CHAT is red, FILE_TRANSFER is orange, etc.).

2.1 Controlling for Uncertainty

Our dashboard has two confidence sliders (Fig. 2) that operators can adjust, which are the primary toggles for dynamic uncertainty quantification filtering for the entire dashboard. Each slider sets a minimum threshold from 0% to 100%, and predicted samples that do not meet both thresholds are reclassified generically

as “OTHER”.

Figure 2: In-Distribution and Class Confidence Sliders

The first In-Distribution slider lets operators set a minimum in-vs-out of distribution confidence, such that samples that are out of distribution are considered OTHER. For example, if the user sets the In-Distribution Confidence slider to 65% but the model is only 64% confident that the sample was in-distribution, the prediction would be reclassified as OTHER. The Class Confidence slider lets operators set a minimum predictive confidence, such that predicted labels that are not confident enough are considered OTHER. Labels are only shown in the dashboard for data that clears both thresholds. Setting high thresholds reduces false positives and gives operators confidence in the results of the model. A large number of out-of-distribution results would indicate that the model needs more training data of that kind, potentially under an entirely new label. Setting low slider thresholds, conversely, reduces false negatives and can be useful for targeted investigation on rare traffic.

2.2 Dashboard Visualizations

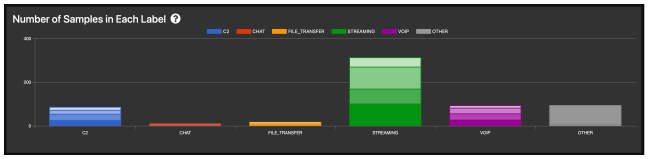

Figure 3: Aggregate Bar Chart

The first visualization is an aggregate bar chart (Fig. 3), in which each bar shows how many samples were predicted to be the given label to provide an overview for the entire dataset. Each bar is further subdivided into darker and lighter color shades to indicate whether the model believes the samples to be in (darker) or out (lighter) of distribution [4]. With this in-vs-out of distribution breakdown, operators can identify data points that their model has not seen before, and thus filter out into OTHER. The side-by-side screenshots in Fig. 1 demonstrate the In-Distribution Confidence filtering capability; Class Confidence behaves similarly.

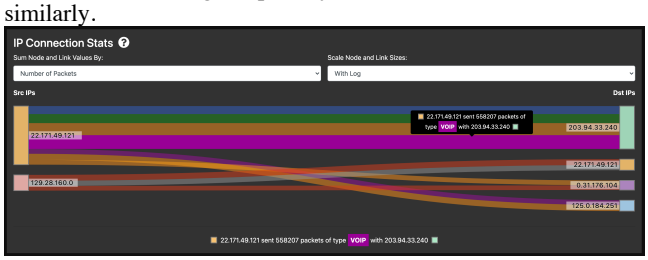

Figure 4: Network Connection Graph

The second view is a connections Sankey diagram (Fig. 4), which has two sides – source IP Address and destination IP Address – with color-coded lines representing connections between the source and destination IPs. The lines are sized by the number of connections, the number of packets, or the total data size and help operators see what actions certain IP addresses were conducting and the bandwidth consumed. This can be helpful for identifying suspicious behavior on a per-connection or IP basis, for example an unauthorized IP or application behavior uncharacteristic to a particular IP.

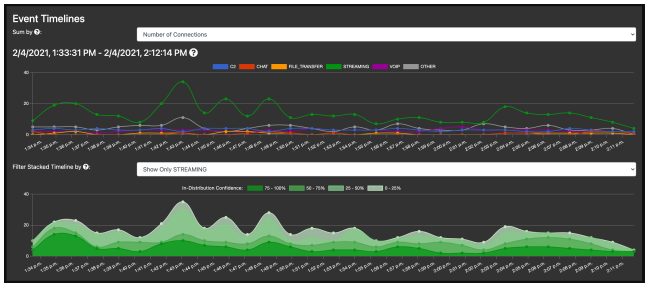

Figure 5: Timeline Visualizations

The final view is a set of timelines (Fig. 5) that visualize the number of connections, the number of packets, or the total data size of the network traffic over time. One timeline overlays the different labels on top of each other which helps operators identify dominant traffic types at different points in time or application workflows. The second timeline displays the same data, but stacks the labels one on top of the other, which betters displays aggregate changes in behavior over time. Moreover, this stacked timeline can be filtered to show, for a given label, the in-vs-out of distribution breakdown over time. These timeline visualizations can be useful for showing temporally related behavior – for example, some initial C2 traffic that triggered FILE_TRANSFER traffic – or a cluster of out of distribution traffic that should be investigated.

In addition to these visualizations, the dashboard displays all the sample data in tabular form, and allows users to completely filter out traffic by the predicted label, IP addresses, or ports, as well as change the color scheme to a colorblind-friendly palette [3].

3 Conclusion

With our dashboard, users can make informed decisions instead of blindly trusting their ML models. Our prototype dynamically moves uncertain predictions into an “OTHER” category. Through the In-Distribution slider and color shading, we help operators identify previously unseen data on which they need to retrain their model to address epistemic uncertainty. Through the Class Confidence slider, we enable users to trust the outputs of their models by filtering out unconfidently classified samples to address aleatoric uncertainty. Moreover, the various visualizations present data in aggregate, network flow, and temporal views, giving operators nuanced SA that can help them determine further courses of action. Giving operators the ability to visually quantify uncertainty can help operators make better decisions with their ML models.

References

[1] 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 Commun. Surv. Tutorials*, vol. 21, no. 2, pp. 1988–2014, 2019, doi: 10.1109/COMST.2018.2883147.

[2] E. Hüllermeier and W. Waegeman, “Aleatoric and epistemic uncertainty in machine learning: an introduction to concepts and methods,” *Mach Learn*, vol. 110, no. 3, pp. 457–506, Mar. 2021, doi: 10.1007/s10994-021-05946-3.

[3] G.-P. Bonneau *et al.*, “Overview and State-of-the-Art of Uncertainty Visualization,” in *Scientific Visualization*, C. D. Hansen, M. Chen, C. R. Johnson, A. E. Kaufman, and H. Hagen, Eds. London: Springer London, 2014, pp. 3–27. doi: 10.1007/978-1-4471-6497-5_1.

[4] S. Guo *et al.*, “Visualizing Uncertainty and Alternatives in Event Sequence Predictions,” in *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, Glasgow Scotland Uk, May 2019, pp. 1–12. doi: 10.1145/3290605.3300803