# Interactive, Visual Uncertainty Quantification

## for Encrypted Network Traffic Situation Awareness

Harry X. Li, Allan B. Wollaber

### Problem

Machine learning (ML) models for situation awareness (SA) in encrypted network traffic often fail to communicate the confidence of their predictions

### Approach

Created a visualization dashboard for encrypted network traffic labels that dynamically adapts to account for uncertainty

### Training and Testing

- Users train or upload an ML model to have calibrated uncertainties via our prototype
- Provide the user a dynamic confusion matrix and other metrics

### Uncertainty Quantification Dashboard

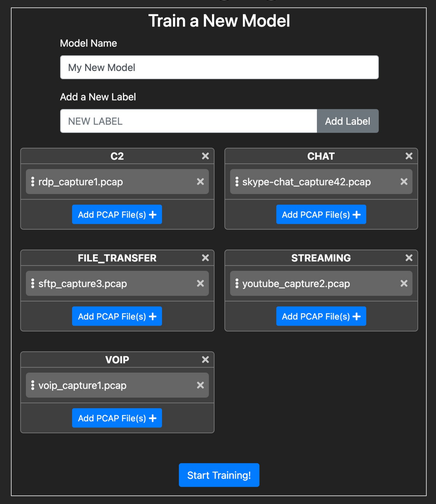
- Users choose ML model, upload unlabeled PCAP
- PCAP is cut up by connection and into time slices
- Samples are assigned "In-Distribution" and "Class Confidence" scores
- In Distribution scores encode *epistemic uncertainty.* 100% (0%) means that the ML model has (has not) seen that kind of data during training
- Class Confidence scores encode *aleatoric uncertainty.* 100% (0%) means that the label prediction is highly (minimally) certain
- The entire dashboard responds to these sliders

### Impact

- Our dashboard enables tuning prediction uncertainty for low- and high-confidence use cases
- Aggregate, network flow, and temporal visualizations give 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

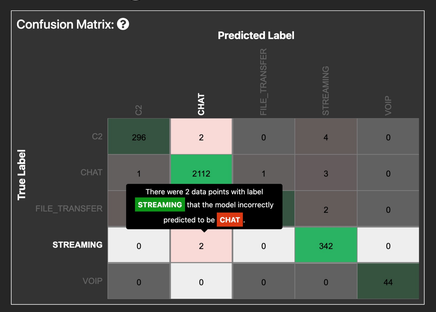
## Training and Testing

### Training Page




- Add/Edit/Delete custom labels
- Assign PCAP files as training data to each label
- Train the new custom model
- When model finishes training
  - View evaluation metrics
  - Make inferences on unseen data

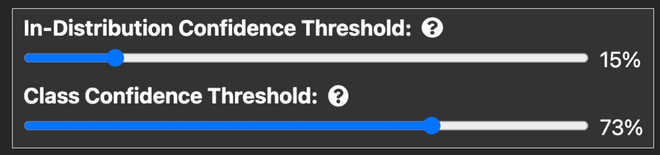
### Testing – Confusion Matrix




- Example of the “usual” model of uncertainty, which only applies to known labels
- Part of a Model Summary Page that includes metrics about training data and validation test performance
- Visualizes class labeling performance of ML model during validation
- Ideally, a model will predict every class perfectly and achieve a perfect score along the green diagonal
- Red mispredictions represent times that the model confused two labels
- High confusion between labels may require more training data or merging labels

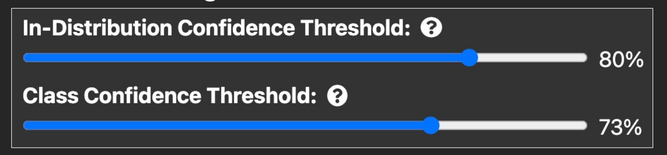
## Uncertainty Quantification Dashboard – Confidence Sliders and Visualizations

### Low Confidence Threshold



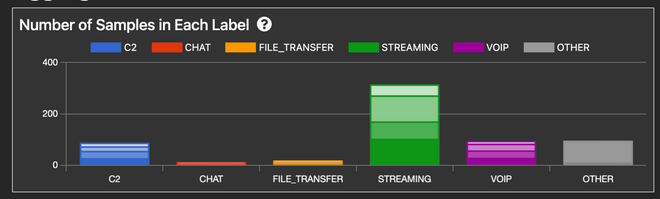

The In-Distribution Confidence slider is set to a lower threshold (15%), allowing the model to make predictions in order to provide some SA.

### High Confidence Threshold

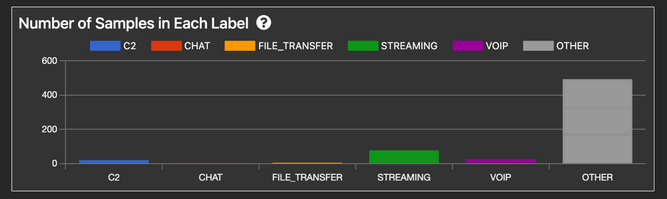



The In-Distribution Confidence slider is set to a higher threshold (80%), letting the operator be more confident in the model’s predictions.

### Aggregate Bar Chart – Shows Overall Classifications of PCAP Data

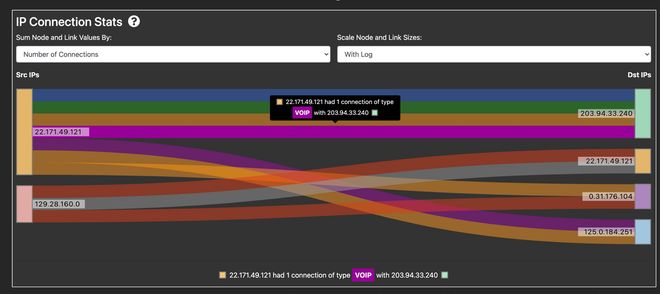


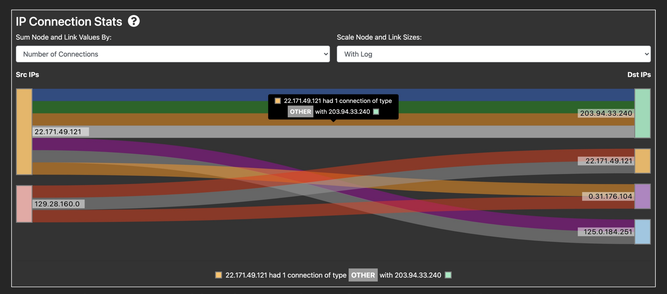

Each bar is subdivided into differently shaded sections. A lighter or darker shade indicates a lower or higher In-Distribution confidence, respectively.




Any lightly shaded sections that do not meet the minimum In-Distribution threshold are moved to the gray OTHER category.

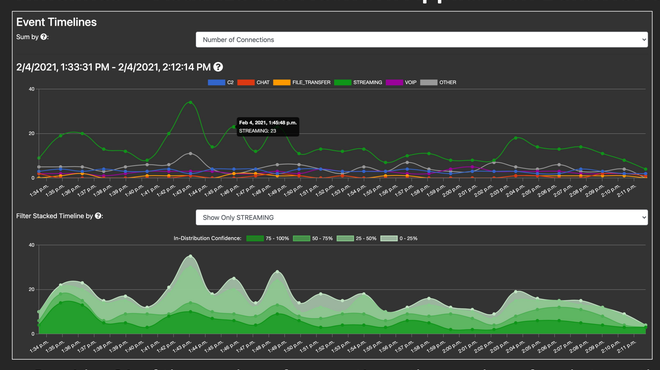
### Network Connections Sankey – Shows Connections Between Source and Destination IP Addresses

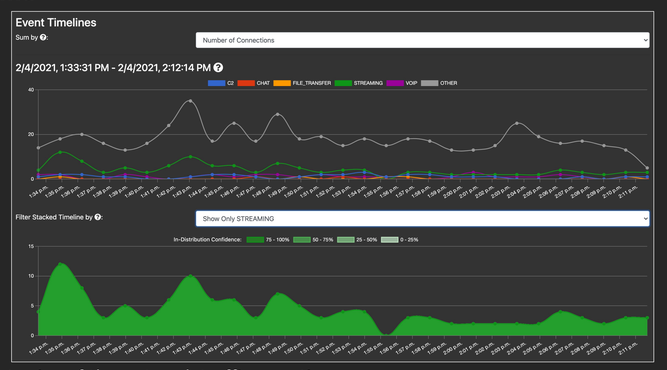



- Lines represent connections between the IP addresses, color coded by application type. They are sized by the number of connections, the number of packets, or the total data size and help operators see applications that certain IP addresses were using and the bandwidth consumed
- Useful for identifying suspicious behavior on a per-connection or IP basis, e.g. application behavior uncharacteristic to a particular IP

### Timelines – Visualize the Network Application Behavior Over Time




- Provides SA of the number of connections, the number of packets, or the total data size of the network traffic over time
- Top timeline overlays the labels on top of each other which helps operators identify dominant traffic types at different points in time or application workflows (e.g. C2 that triggered FILE_TRANSFER)
- Bottom timeline displays the same data, but stacks the labels one on top of the other, which betters displays aggregate changes in behavior over time. It can be filtered to show, for a given label, the in-vs-out of distribution breakdown over time

**Lincoln Laboratory POCs: Harry X. Li <Harry.Li@ll.mit.edu>, Allan B. Wollaber <Allan.Wollaber@ll.mit.edu>**



LINCOLN LABORATORY
MASSACHUSETTS INSTITUTE OF TECHNOLOGY