Incremental Learning for Mobile Encrypted Traffic Classification

Yige Chen∗†, Tianning Zang∗†, Yongzheng Zhang∗†, Yuan Zhou ‡, Linshu Ouyang∗†, Peng Yang‡

∗ Institute of Information Engineering, Chinese Academy of Sciences
† School of Cyber Security, University of Chinese Academy of Sciences
‡ National Computer Network Emergency Response Technical Team/Coordination Center of China
Closed-world mobile encrypted classification

• Classify encrypted traffic into its belonging application
Closed-world mobile encrypted classification

• Classify encrypted traffic into its belonging application

Open-world mobile encrypted classification

• Breaks the closed-world assumption
Closed-world mobile encrypted classification

• Classify encrypted traffic into its belonging application

Open-world mobile encrypted classification

• Breaks the closed-world assumption
• Deal with the unseen applications
Closed-world Encrypted Traffic Classification

Mobile Encrypted Traffic

Classifier

Existing Applications

<table>
<thead>
<tr>
<th>Existing state-of-the-art methods</th>
<th>TPR</th>
<th>FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>SoB [1]</td>
<td>91.22%</td>
<td>0.08%</td>
</tr>
<tr>
<td>MaMPF [2]</td>
<td>96.40%</td>
<td>0.20%</td>
</tr>
<tr>
<td>FS-Net [3]</td>
<td>99.14%</td>
<td>0.05%</td>
</tr>
<tr>
<td>MAAF [4]</td>
<td>98.69% Acc</td>
<td>98.64% F1</td>
</tr>
</tbody>
</table>

Closed-world Encrypted Traffic Classification

Mobile Encrypted Traffic

Existing state-of-the-art methods

<table>
<thead>
<tr>
<th>Method</th>
<th>TPR</th>
<th>FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>SoB [1]</td>
<td>91.22%</td>
<td>0.08%</td>
</tr>
<tr>
<td>MaMPF [2]</td>
<td>96.40%</td>
<td>0.20%</td>
</tr>
<tr>
<td>FS-Net [3]</td>
<td>99.14%</td>
<td>0.05%</td>
</tr>
<tr>
<td>MAAF [4]</td>
<td>98.69% Acc</td>
<td>98.64% F1</td>
</tr>
</tbody>
</table>


Closed-world Encrypted Traffic Classification

Existing state-of-the-art methods

<table>
<thead>
<tr>
<th>Method</th>
<th>TPR</th>
<th>FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>SoB [1]</td>
<td>91.22%</td>
<td>0.08%</td>
</tr>
<tr>
<td>MaMPF [2]</td>
<td>96.40%</td>
<td>0.20%</td>
</tr>
<tr>
<td>FS-Net [3]</td>
<td>99.14%</td>
<td>0.05%</td>
</tr>
<tr>
<td>MAAF [4]</td>
<td>98.69% Acc</td>
<td>98.64% F1</td>
</tr>
</tbody>
</table>

Closed-world Encrypted Traffic Classification

Mobile Encrypted Traffic

Classifier

Existing Applications

<table>
<thead>
<tr>
<th>Existing state-of-the-art methods</th>
<th>SoB [1]</th>
<th>91.22% TPR</th>
<th>0.08% FPR</th>
</tr>
</thead>
<tbody>
<tr>
<td>MaMPF [2]</td>
<td>96.40% TPR</td>
<td>0.20% FPR</td>
<td></td>
</tr>
<tr>
<td>FS-Net [3]</td>
<td>99.14% TPR</td>
<td>0.05% FPR</td>
<td></td>
</tr>
<tr>
<td>MAAF [4]</td>
<td>98.69% Acc</td>
<td>98.64% F1</td>
<td></td>
</tr>
</tbody>
</table>

Open-world Encrypted Traffic Classification

Mobile Encrypted Traffic

Initial Classifier

Existing Applications
Open-world Encrypted Traffic Classification

Existing Applications

When we find new applications
Zhang et al. [5] [6]

More Applications


Open-world Encrypted Traffic Classification

Mobile Encrypted Traffic

Initial Classifier

Updated Classifier

Existing Applications

When we find new applications

More Applications
Open-world Encrypted Traffic Classification

Mobile Encrypted Traffic → Initial Classifier → Updated Classifier

 Incremental Learning? 

Existing Applications

When we find new applications

More Applications
Naive Incremental Learning Methods

1. Retraining the updated classifier from scratch
   • Considerable training time and effort
   • Expansion of the dataset
Naive Incremental Learning Methods

1. Retraining the updated classifier from scratch
   • Considerable training time and effort
   • Expansion of the dataset

2. Fine-tuning the existing classifier
   • catastrophic forgetting problem
Incremental Learning based on (OvR) Strategy
Incremental Learning based on (OvR) Strategy

One vs Rest Strategy

- \( n \) binary classifiers. The classifier \( h_i \) correspond to \( i^{th} \) mobile application.
Incremental Learning based on (OvR) Strategy

One vs Rest Strategy

- \( n \) binary classifiers. The classifier \( h_i \) correspond to \( i^{th} \) mobile application.
- The binary classifier \( h_i \) considers \( i^{th} \) application as positive while other applications as negative.
Incremental Learning based on (OvR) Strategy

One vs Rest Strategy

- \( n \) binary classifiers. The classifier \( h_i \) correspond to \( i^{th} \) mobile application.
- The binary classifier \( h_i \) considers \( i^{th} \) application as positive while other applications as negative.
- The system integrates all binary classifiers to make classification.
Incremental Learning based on (OvR) Strategy

- Collect the dataset of new applications.
Incremental Learning based on (OvR) Strategy

Incremental Learning

• Collect the dataset of new applications.
• Build extra new binary classifiers for the new applications.
Incremental Learning based on (OvR) Strategy

Incremental Learning

- Collect the dataset of new applications.
- Build extra new binary classifiers for the new applications.
- Retrain the outdated classifier that accept more than the retraining threshold $\tau$ of the new applications’ traffic
Binary Classifier

1. A neural network-based implementation.

Neural Binary Classification Network
Binary Classifier

1. A neural network-based implementation.

2. Take the first $k$ packet lengths of flows as classifier input.
Binary Classifier

1. A neural network-based Implementation.
2. Take the first k packet lengths of flows as classifier input.
3. Use Gated Recurrent Unit (GRU) to process the sequence inputs.

Neural Binary Classification Network
Binary Classifier

1. A neural network-based implementation.

2. Take the first k packet lengths of flows as classifier input.

3. Use Gated Recurrent Unit (GRU) to process the sequence inputs.

4. Adopt a sigmoid function to normalize classification probability $\in (0-1)$.

Neural Binary Classification Network
Sample Selection

1. Maintain the size of dataset when adding new applications.
Sample Selection

1. Maintain the size of dataset when adding new applications.

2. Select new samples and remove leftover samples through herding selection.
Sample Selection

1. Maintain the size of dataset when adding new applications.

2. Select new samples and remove leftover samples through herding selection.

3. Sequentially selects samples that keeps its vector average nearest to the original vector average.
Evaluation Dataset

- A manually collected dataset provided by MAAF [1].
- 77,278 real-world encrypted flows of 16 popular mobile applications.

Analysis of the Retraining Threshold

Performance Evaluation on Different Retraining Thresholds
Analysis of the Retraining Threshold

• Both the F1-scores and the number of retrained classifiers declines with the retraining threshold.
Analysis of the Retraining Threshold

- Both the F1-scores and the number of retrained classifiers declines with the retraining threshold.
Analysis of the Retraining Threshold

• Both the F1-scores and the number of retrained classifiers declines with the retraining threshold.

• When the retraining threshold increases from 0 to 0.1, the number of retrained classifiers decreases sharply while the classifier only loses a little F1-scores.
Analysis of the Sample Selection

Comparison Results on Different Sample Numbers
Analysis of the Sample Selection

Comparison Results on Different Sample Numbers
Analysis of the Sample Selection

• The F1-score of both IL-Herding and IL-Random increases.
Analysis of the Sample Selection

• The F1-score of both IL-Herding and IL-Random increases.
• The IL-Herding gradually approaches IL-Base with the enrichment of reserved samples.
Analysis of the Sample Selection

- The F1-score of both IL-Herding and IL-Random increases.
- The IL-Herding gradually approaches IL-Base with the enrichment of reserved samples.
- The IL-Herding shows overall higher classification accuracy than the IL-Random.

Comparison Results on Different Sample Numbers
Analysis of the Number of Applications

Comparison Results on Different Numbers of Applications
Analysis of the Number of Applications

Comparison Results on Different Numbers of Applications
Analysis of the Number of Applications

• The F1-scores of both the IL-Herding and MC-Herding slowly decrease with the increase of the number of applications.
Analysis of the Number of Applications

• The F1-scores of both the IL-Herding and MC-Herding slowly decrease with the increase of the number of applications.

• The number of retrained classifiers generally declines from 2.0 to 1.0 with the increase in the number of applications.

Comparison Results on Different Numbers of Applications
• For more details, please contact chenyige@iie.ac.cn

• Questions & Answers