PTN-IDS: Prototypical Network Solution for the Few-shot Detection in Intrusion Detection Systems

Nadia Niknami, Vahid Mahzoon, and Jie Wu

Dept. of Computer and Info. Sciences

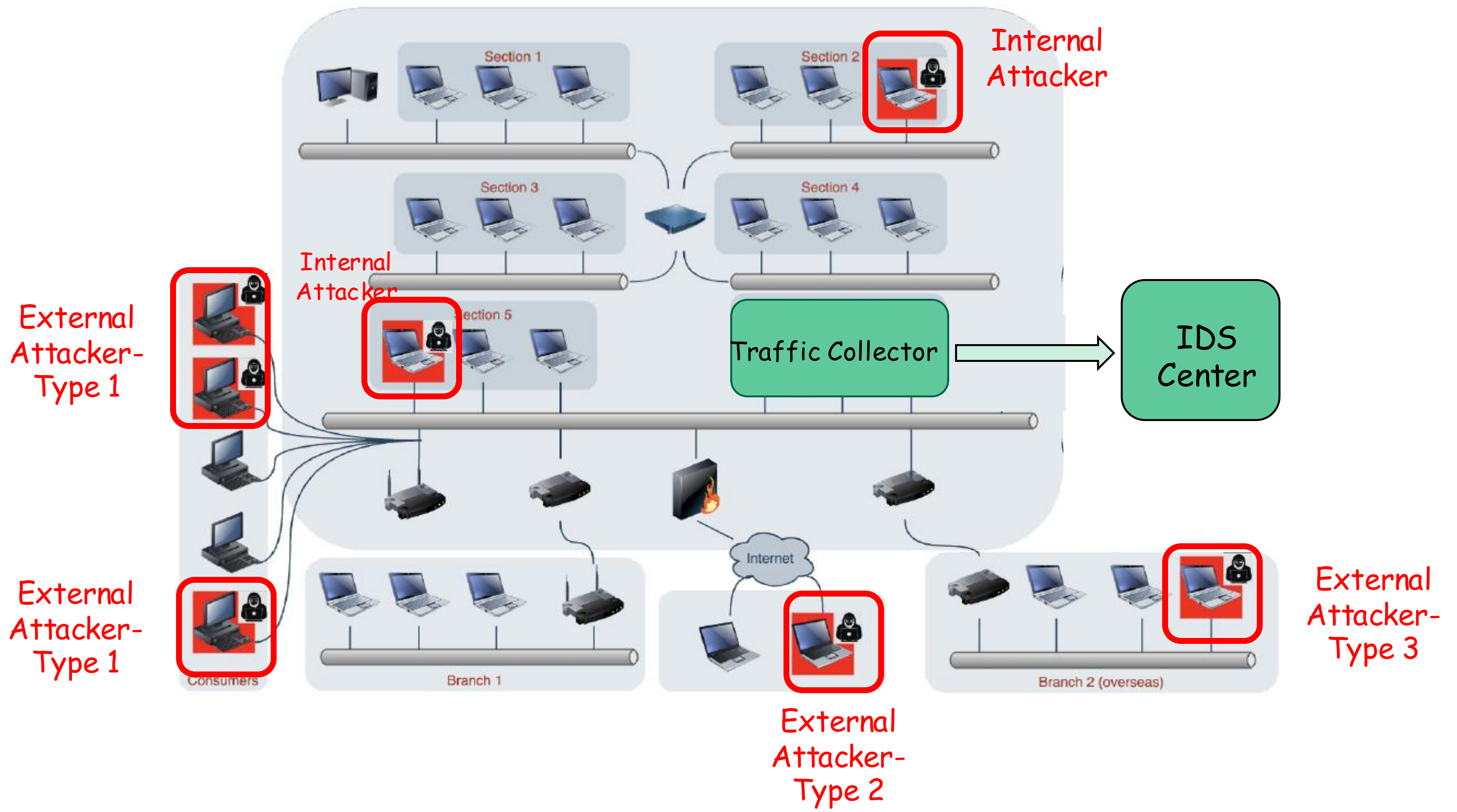
Temple University

Outline

- Intrusion Detection Systems (IDS)
- Introduction to Few-Short Learning (FSL) in Meta Learning
- Introduction to Prototypical Networks (PTN)
- The Proposed Approach: PTN-IDS
- Experiment Results of PTD-IDS
- Conclusions

1. Intrusion Detection Systems (IDS)

- IDS is a network security tool
  - monitors network traffic and devices for known malicious activity, suspicious activity, or security policy violations.



Problems of Existing IDS

Zero-day Attack Detection:

- A Zero-day attack exploits vulnerabilities for which no prior training data exists.
- **Challenge:** Traditional IDS struggle to detect such attacks without prior knowledge.

Domain Shift:

- Differences between training and testing data distributions.
- **Challenge:** Models trained on one dataset often fail to generalize to different datasets due to domain shifts.

Using Few-Shot Learning (FSL) and Prototypical Network (PTN) help IDS to detect attacks with minimal data and adapt to varying data distributions.

2. Few-Shot Learning in Meta-Learning

Goal: train the model to accurately classify new, unseen examples even when only a small number of examples are given during training.

- Data is split into:
  - Support set (used for training on that task)
  - Query set (used for testing the task)

- The model must classify instances from N classes with K examples each in each task (N-way K-shot)
  - N-way: Support set has N classes
  - K-Shot: Every class has K samples
    - E.g., 5-shot learning is trained with 5 examples per class.

Few-Shot Learning Examples

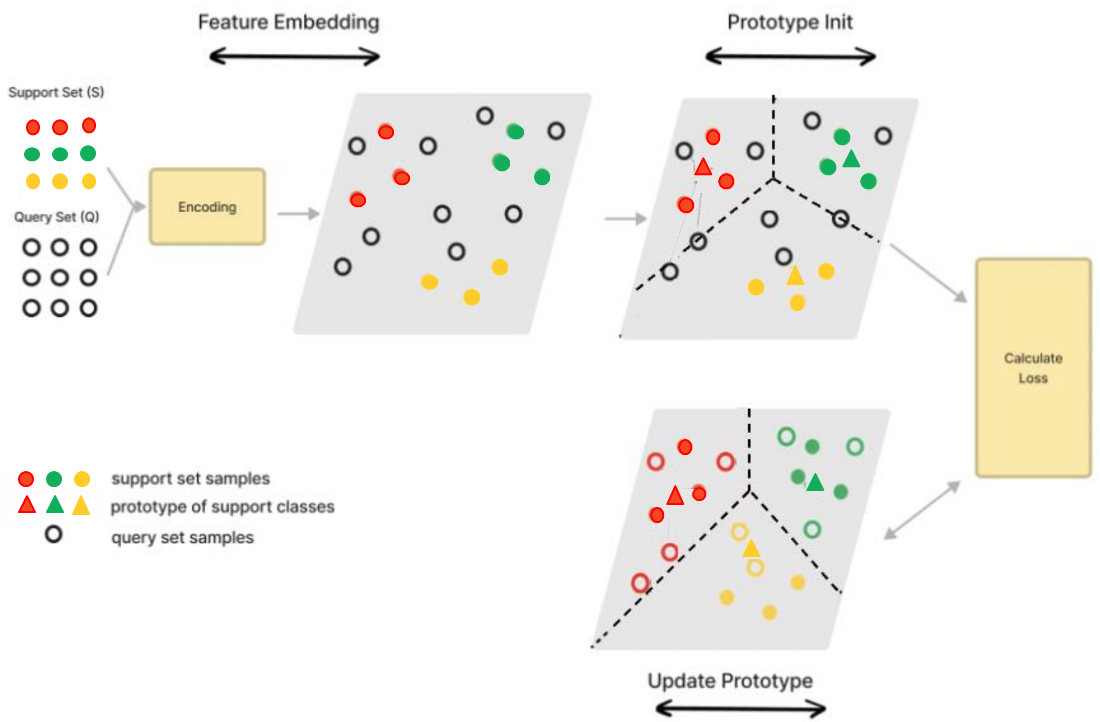
3-way 2-shot

Few-Shot Learning (Cont'd)

- Task
  - A mini-classification problem with N classes and K examples per class.
- Number of tasks
  - Refers to how many distinct learning scenarios the model is exposed to during meta-training.
- Training across many tasks
  - To learn a generalizable representation that allows quick adaptation to new tasks, with very few examples per class.

3. Prototypical Networks (PTN)

- PTN: a metric-based method that computes distances to prototype representations of each class for classification.
  - Smaller distances indicate a higher likelihood
- Feature Embeddings: generating high-dimensional vectors that capture the important features of the input data.



Multi-class Classification Embedding

IDS is reliable if sufficient data is available for all attacks

t-distributed Stochastic Neighbor Embedding (t-SNE)

- Visualizing high-dimensional data by reducing it to 2D or 3D.
- A non-linear dimensionality reduction technique.

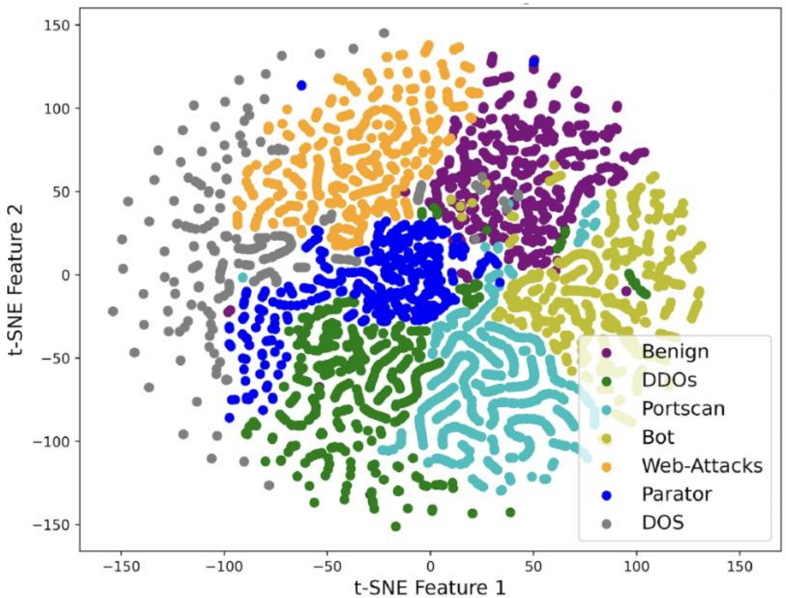
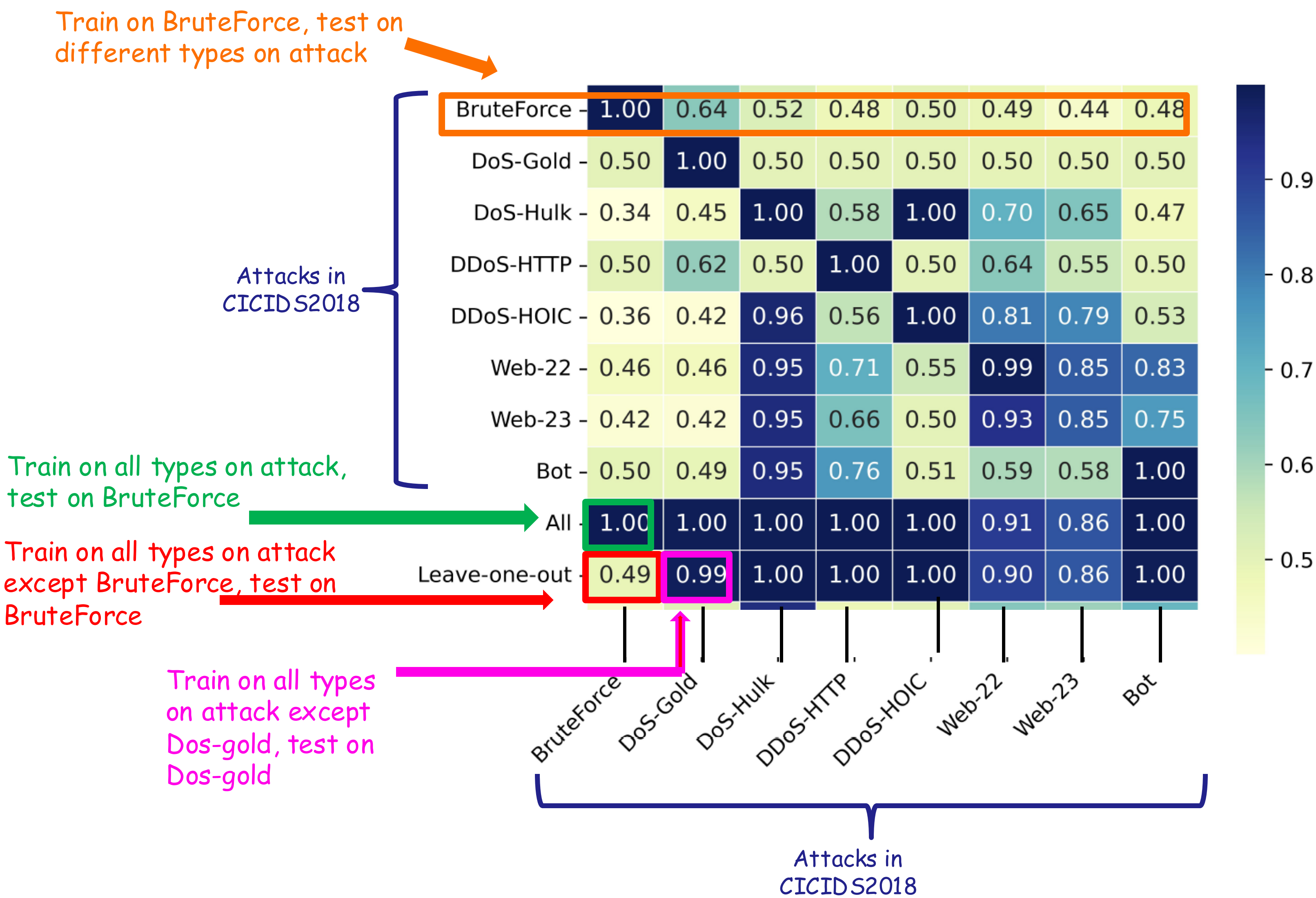
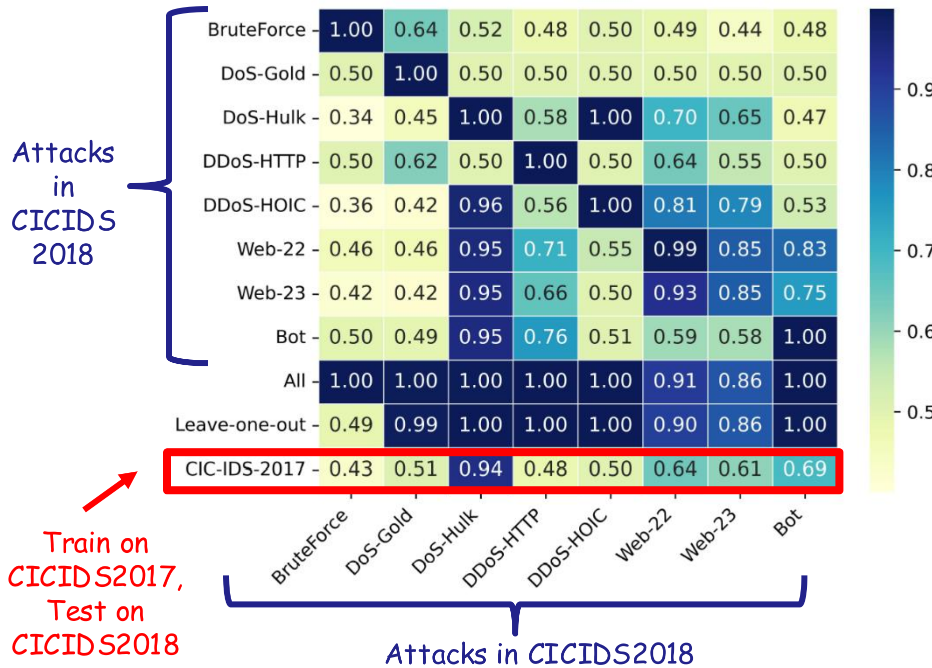


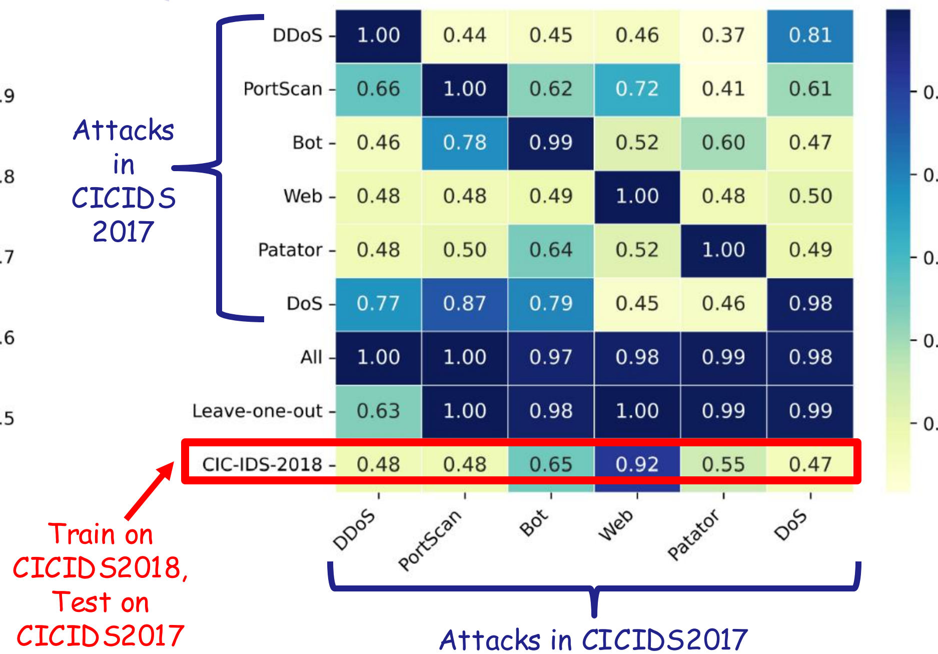
Fig. 2: Multi-class classification embedding.

Issue: Zero-day Attack



Issue: Domain Shift





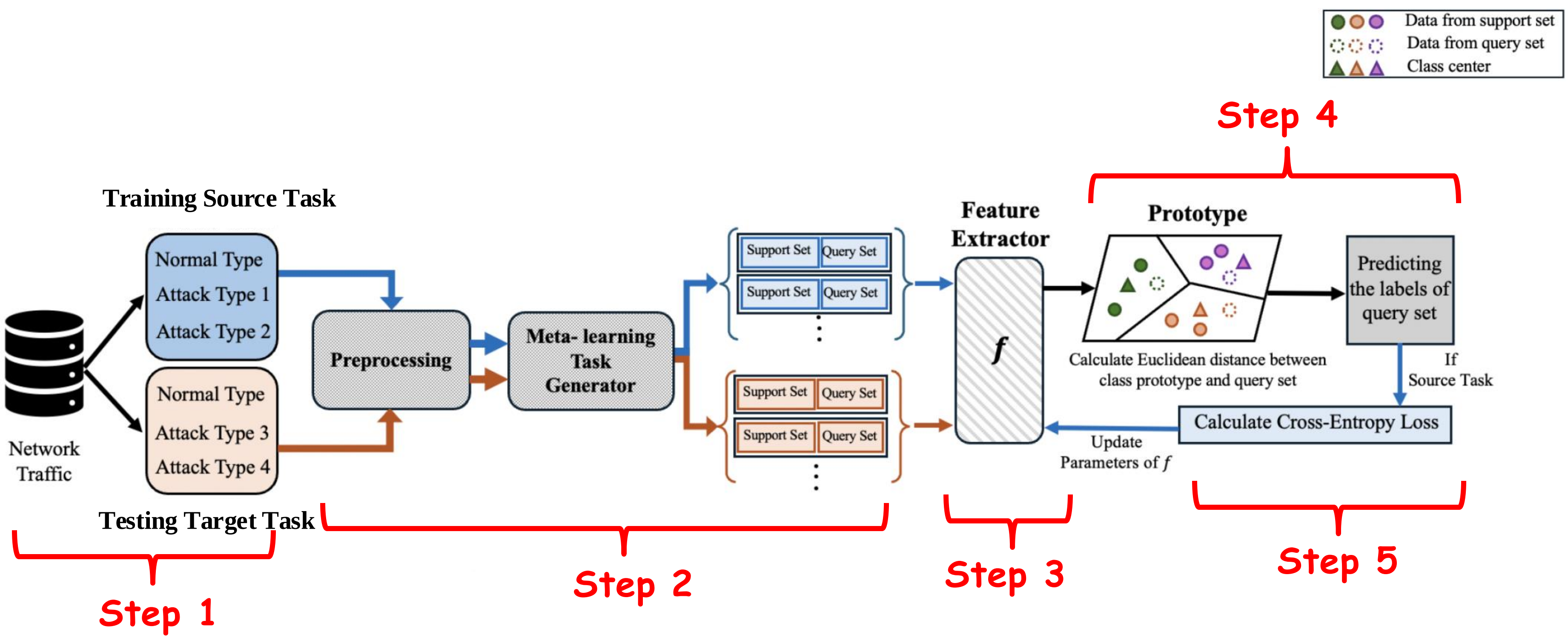
CICIDS2017 dataset:

Simulates real-world network traffic with benign and other attacks including DDoS, DoS, Botnet, PortScan, Patator, and Web Attacks. It contains 80 feature.

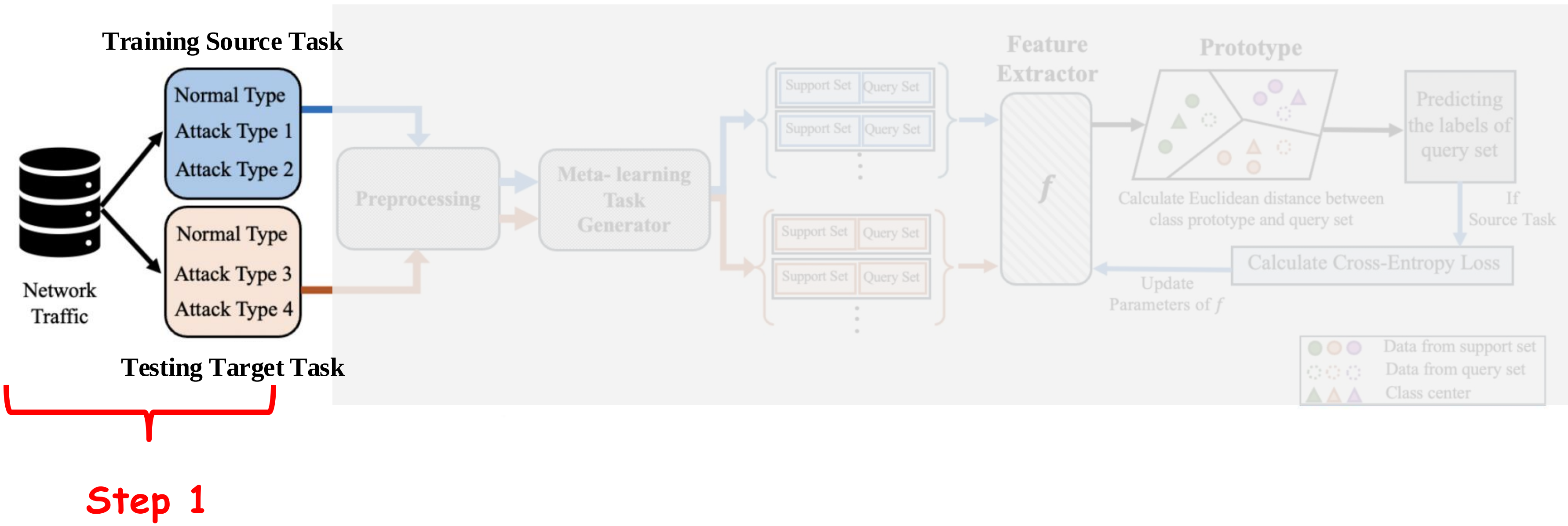
CICIDS2018 dataset:

It enhances the 2017 version, with more attacks, including DoS-Gold, DDoS-HTTP, Brute Force, Botnet, and some new attack types.

4. Proposed PTN-based Intrusion Detection System: PTN-IDS

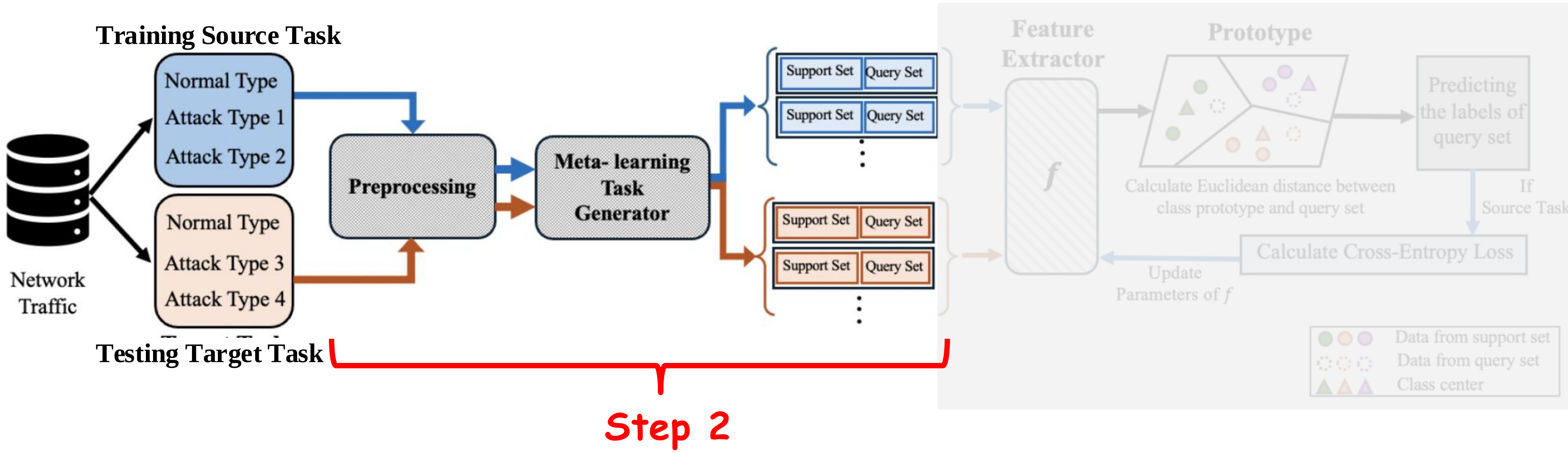


Proposed PTN-IDS (1)



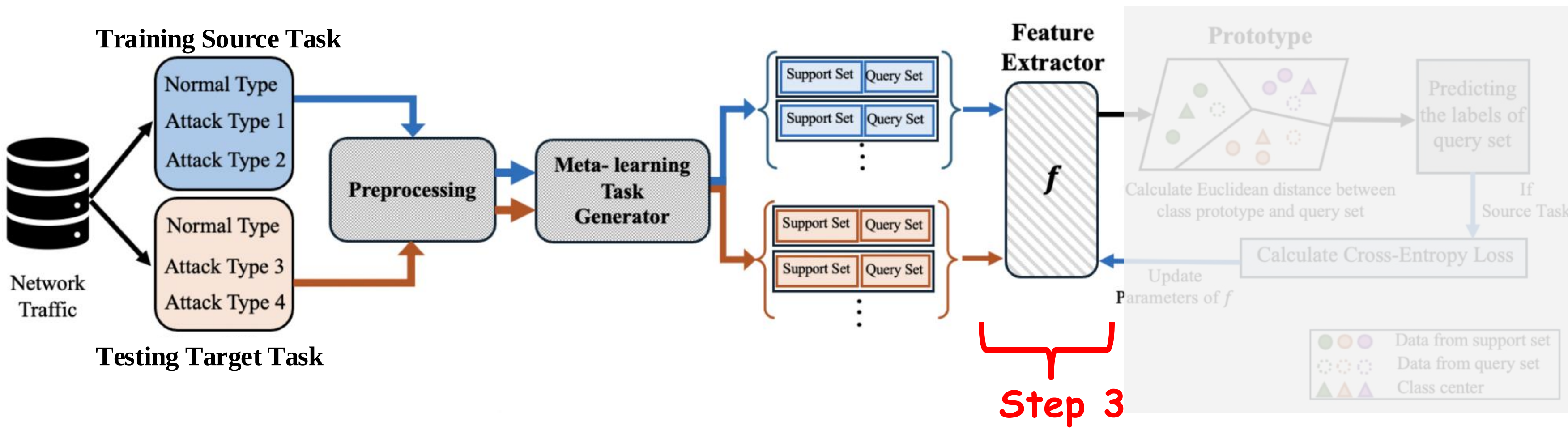
- The network traffic dataset is divided into two distinct tasks:
  - Training Source task
  - Testing Target task.
- There is no overlap in the label spaces.
- Attack types in the source and target tasks are different.

Proposed PTN-IDS (2)



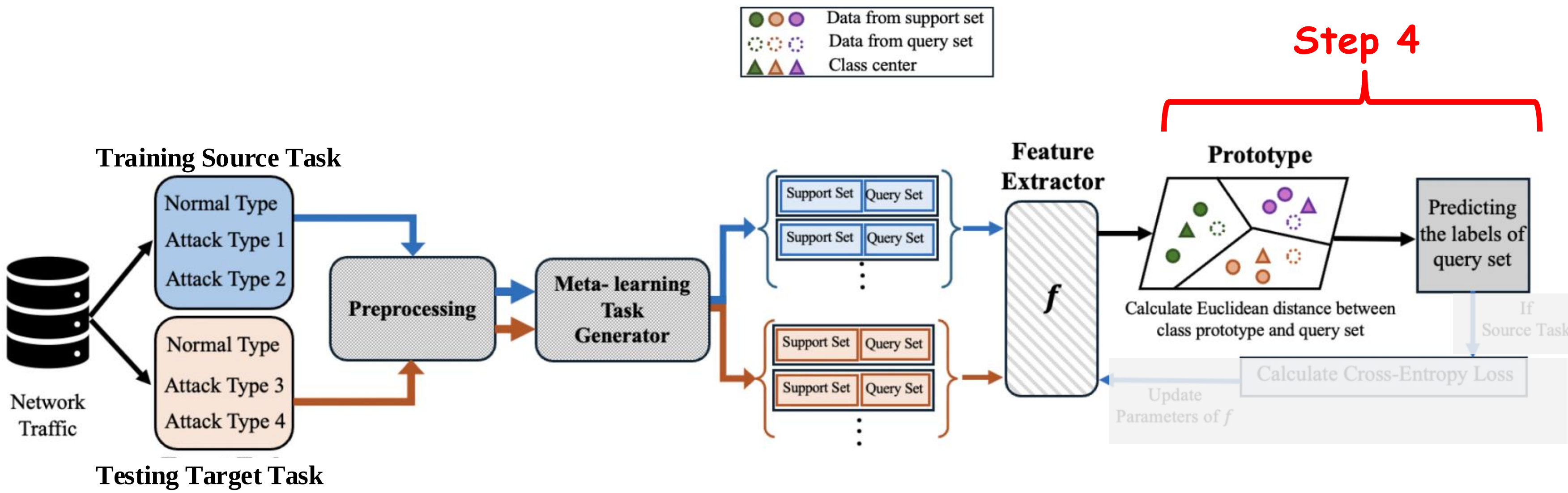
- Preprocessing: cleaning and normalizing data
- Generate support set (labeled examples) and query set (examples used to evaluate the model's performance on the task) in source and target tasks.

Proposed PTN-IDS (3)



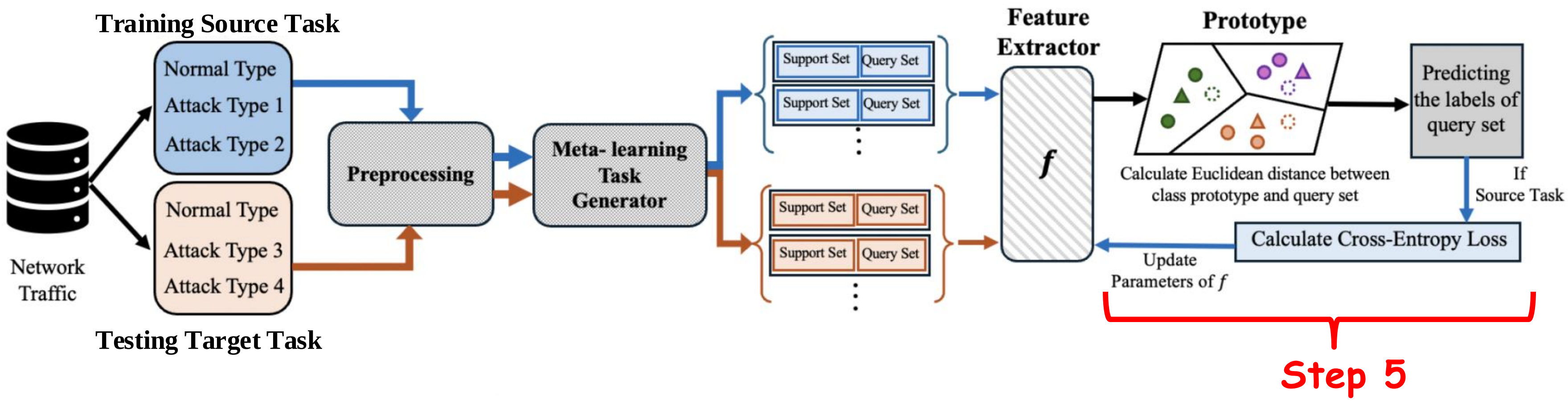
- Feature extractor is a neural network (NN).
- NN processes the raw network traffic data, transforming it into embeddings—high-dimensional vectors that capture the important features of the input data.

Proposed PTN-IDS (4)



- For each class, a prototype is computed by taking the mean vector of the embeddings from the support set.
- Predict label for query set.

Proposed PTN-IDS (5)



- Calculates the cross-entropy loss between the predicted and true labels of the query set.
- Update the parameters of the feature extractor neural network through backpropagation.

5. Experimental Results Different n, k, and Distance Function

TABLE II: Comparison of Baseline and Proposed Method across Different n-values

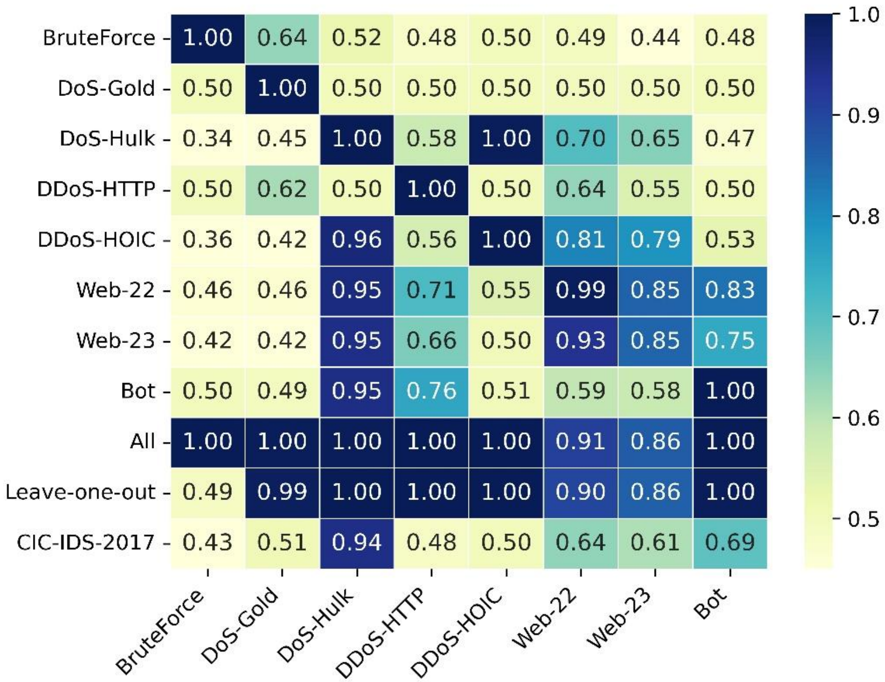
Models	Scenario1: n=1		Scenario2: n=2		Scenario3: n=3	
	Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score
Baseline	0.6271	0.5814	0.5957	0.5488	0.5067	0.4541
1-shot Proposed	0.7918	0.7651	0.7014	0.6797	0.6345	0.5924
5-shot Proposed	0.9102	0.9067	0.8297	0.8232	0.7946	0.7785
10-shot Proposed	0.9312	0.9296	0.8445	0.8370	0.8186	0.8084

TABLE III: Comparison of using Different Distance Function in PTN with 5-shot.

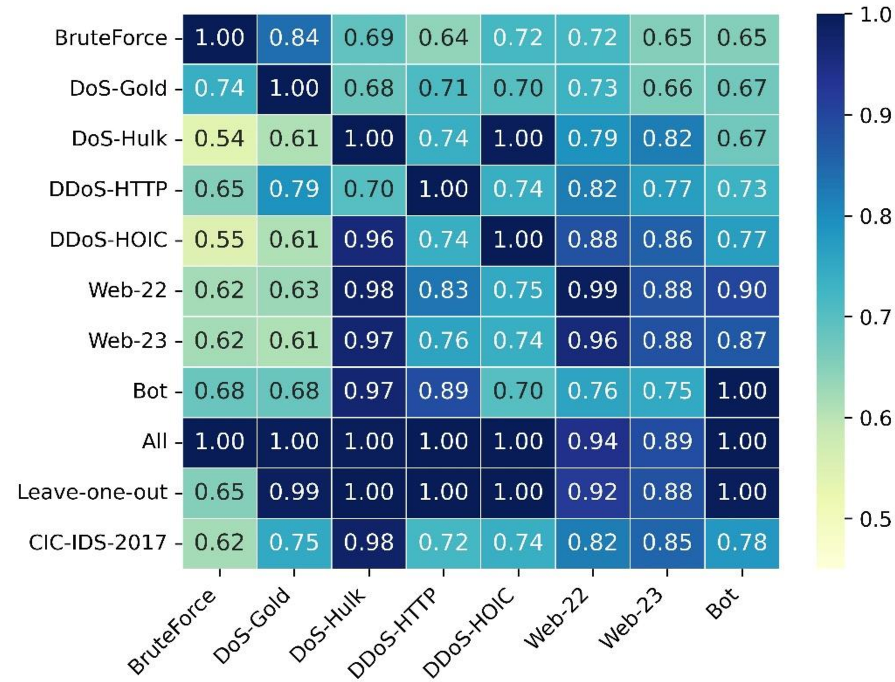
Models	Scenario1: n=1		Scenario2: n=2		Scenario3: n=3	
	Accuracy	F1-score	Accuracy	F1-score	Accuracy	F1-score
Euclidean Distance	0.9102	0.9067	0.8297	0.8232	0.7946	0.7785
Manhattan Distance	0.8860	0.8779	0.8134	0.8035	0.7797	0.7682
Cosine Distance	0.9098	0.9048	0.7285	0.6964	0.7691	0.7560

Scenario 1 :DDoS in the target task
Scenario 2: Web Attack and DoS
Scenario 3 :Web Attack, DoS, and PortScan

Experimental Results on Zero-day Attacks



Zero-day Attack Detection
without FSL and PTN



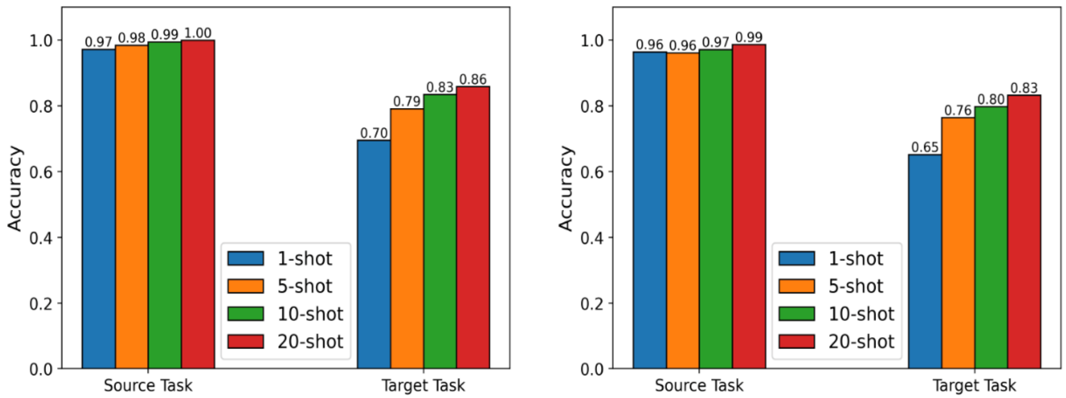
Zero-day Attack Detection
with FSL and PTN

The proposed Model

Experiment Results on Zero-day Attacks

- Scenario 1:
  - Source task is {DDoS, PortScan, Bot}
  - Target task is {Web, BruteForce, DoS}

- Scenario 2:
  - Source task is {Web, BroutForce, DoS}
  - Target task is {DDoS, PortScan, Bot}



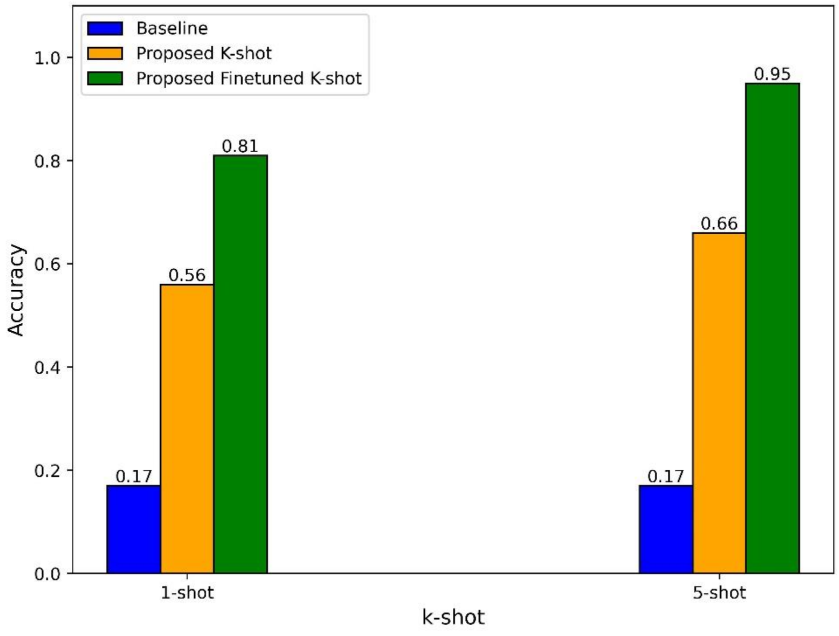
(a) Scenario 1 (b) Scenario 2

Fig. 7: Detecting Zero-day attack in different scenarios.

As k increases, there is a further improvement in classification accuracy.

Experiment Results on Domain Shift

- Fine-tuning is model's adaptation to the target task in the source task.
- Fine-tuning improves the accuracy in the target task.
- With 5 shots, there is an improvement for accuracy of attack detection.



Conclusions

- Effectiveness of FSL and PTN in scenarios with limited labeled data.
- Reach a high accuracy, using 5 samples from each label.
- Classifying Zero-day attacks with high accuracy.
- Adaptability to domain shift between datasets.