RadioNet: Robust Deep-Learning Based Radio Fingerprinting

Haipeng Li[1], Kaustubh Gupta[2], Chenggang Wang[1],

Nirnimesh Ghose[2], Boyang Wang[1]

[1] University of Cincinnati, USA; [2] University of Nebraska-Lincoln;

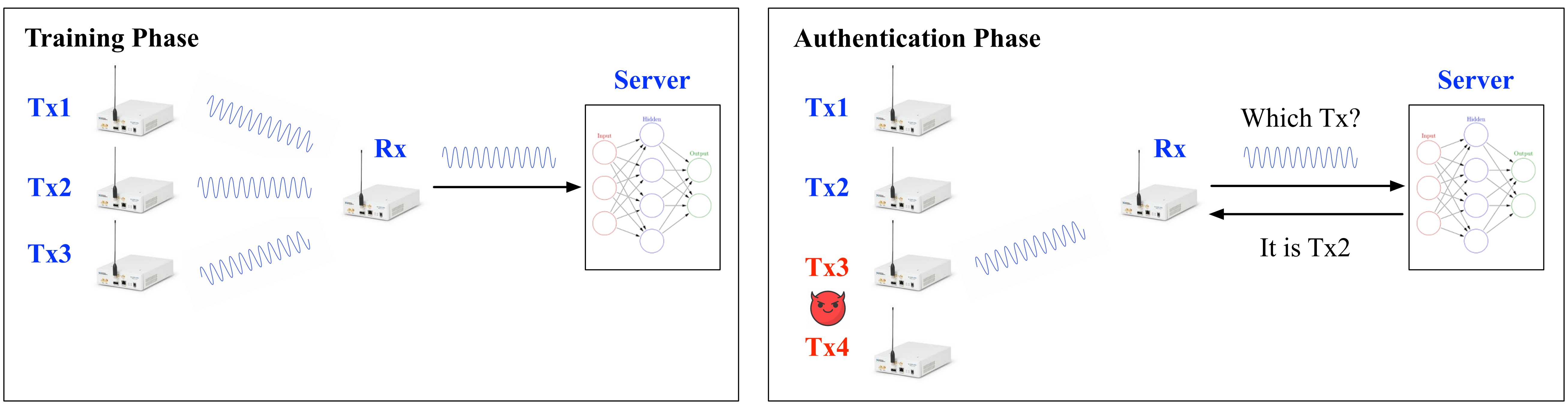
IEEE CNS 2022

University of Cincinnati

University of Nebraska Lincoln

Radio Fingerprinting

Authenticating wireless devices over Radio Frequency (RF) signals at the physical layer.

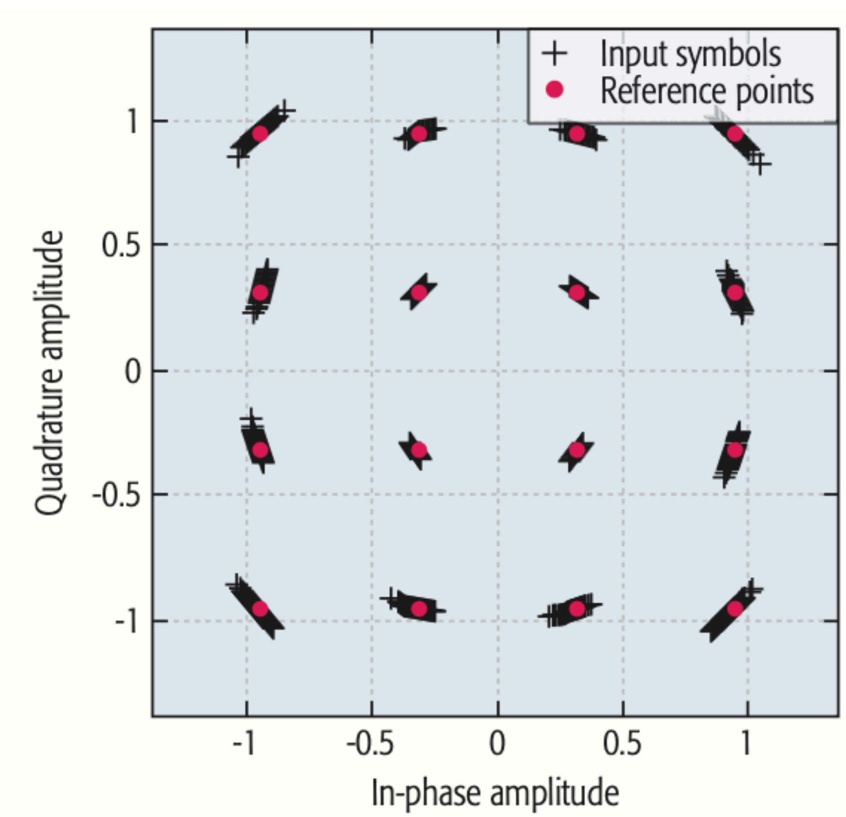


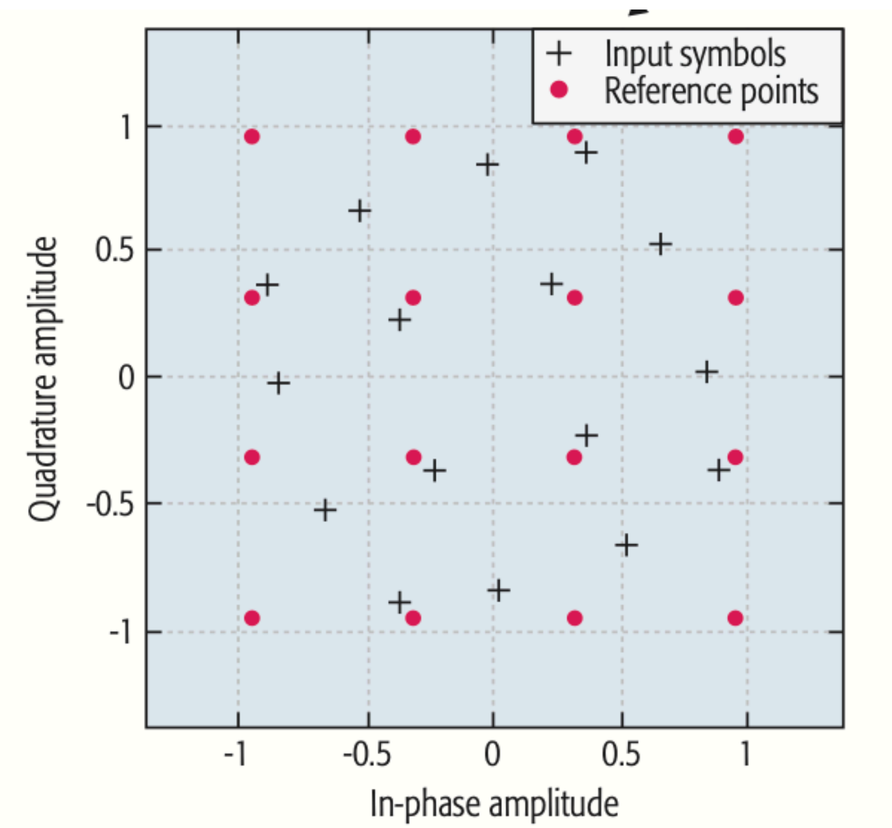
Why Feasible?

- Hardware imperfections (I/Q imbalance, phase noise, nonlinear distortion, etc.) lead to minor shifts in RF signals.

- Each transmitter has unique hardware imperfections

Phase noise





Nonlinear distortion

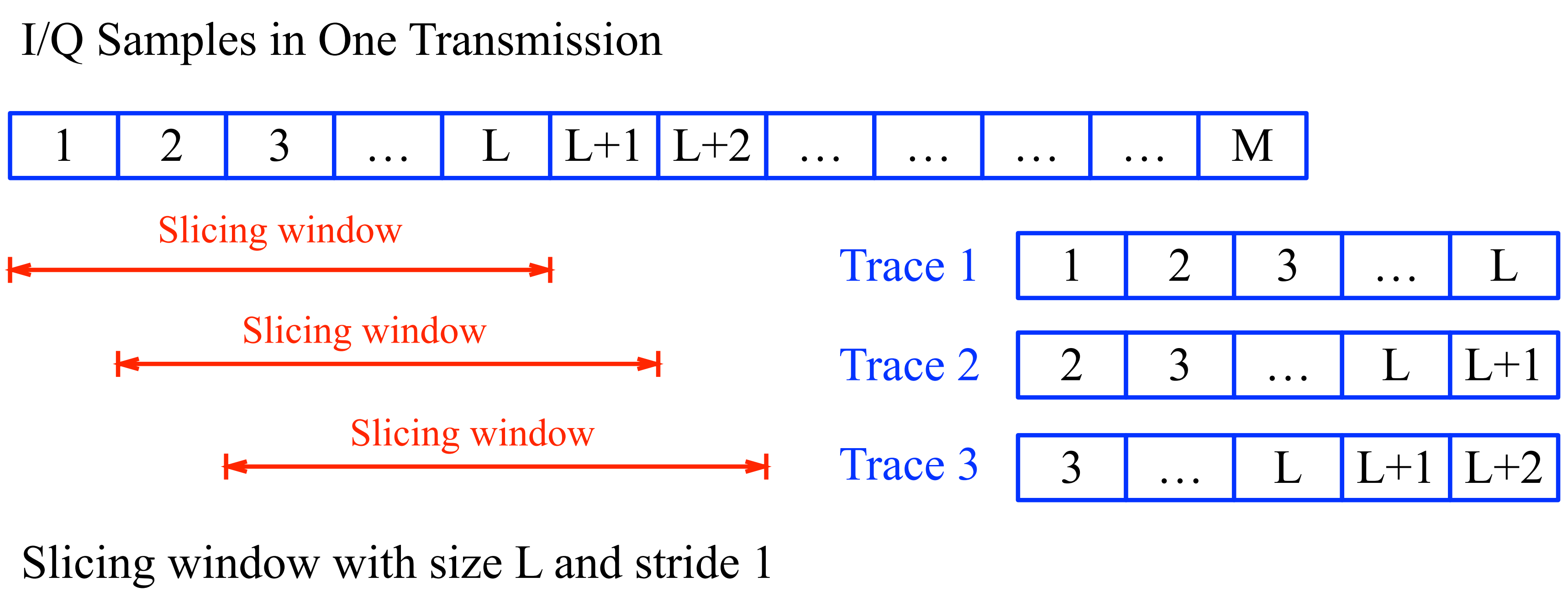
Figures are from S. Riyaz, “Deep learning convolutional Neural Networks for Radio Identification,” IEEE Magazine, Sept., 2018

Limitations in Radio Fingerprinting

- Deep learning achieves high accuracy in same-day
  - IEEE Magazine’18, INFOCOM’19, EuroS&P’20.
  - Same-day: train with Day 1, test with Day 1
- **Significant Performance drop in a cross-day scenario**
  - MobiHoc’19, Globecom’19, INFOCOM’20,
  - Cross-day: train with Day 1, test with Day 2
- Example: 20 transmitters (USRPs), CNN as classifier
  - Same-day accuracy: 99%
  - Cross-day accuracy: 5% (random guess)

Limitations in Radio Fingerprinting

- Slicing windows are often used to pre-process RF signals to inputs of neural networks
- **The selection of parameters in pre-processing has not been rigorously discussed**



Our Contributions

- Improve **robustness** of Radio Fingerprinting from 3 aspects:

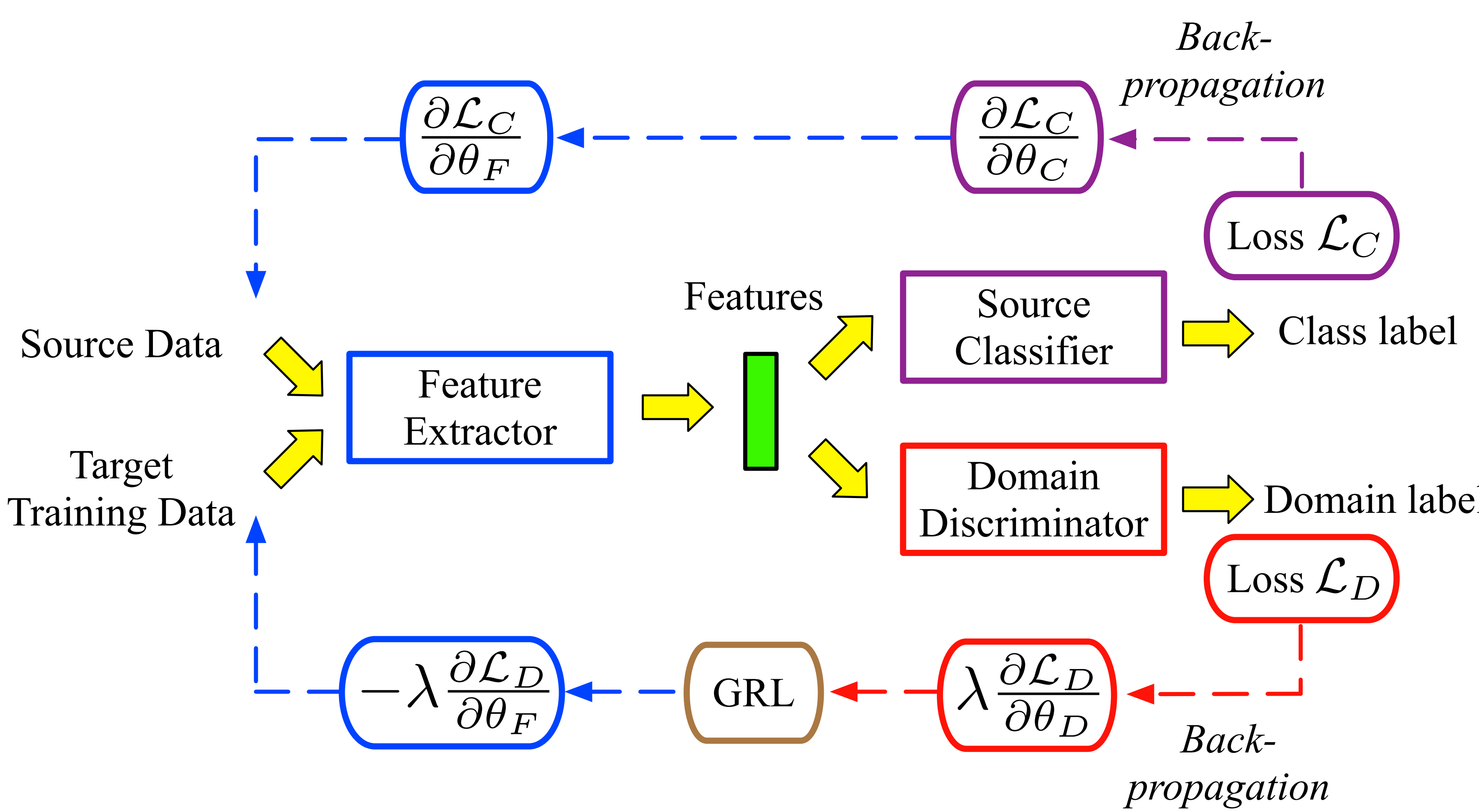
  1. Demonstrate that parameters of pre-processing have significant impacts to accuracy (from extremely high to random guess)

  2. Improve cross-day accuracy with adversarial domain adaptation

  3. Introduce device rank as a more robust metric

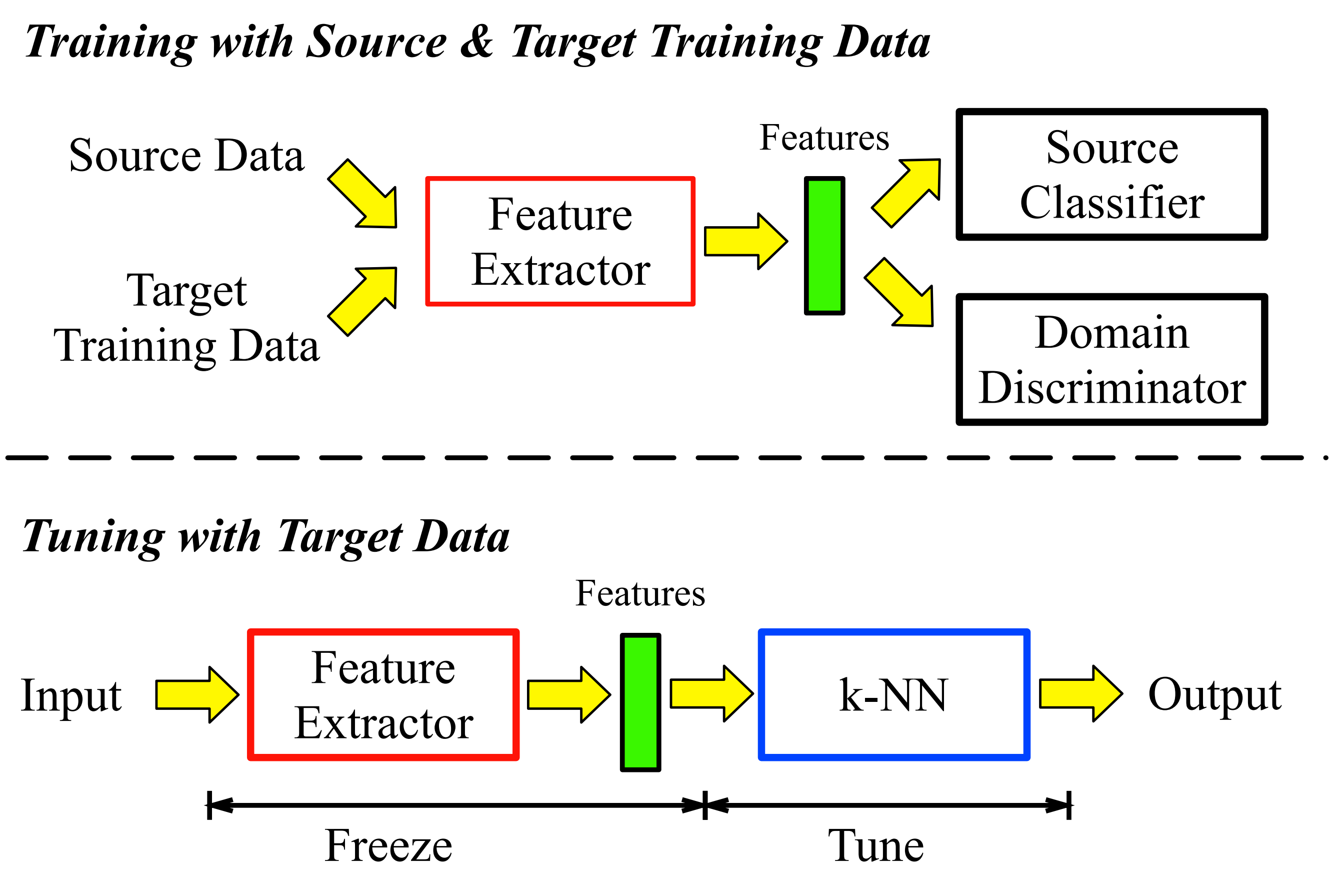
Adversary Domain Adaptation

- Given source data and target data, ADA minimizes the discrepancy between source & target in a feature space



Our Method with ADA

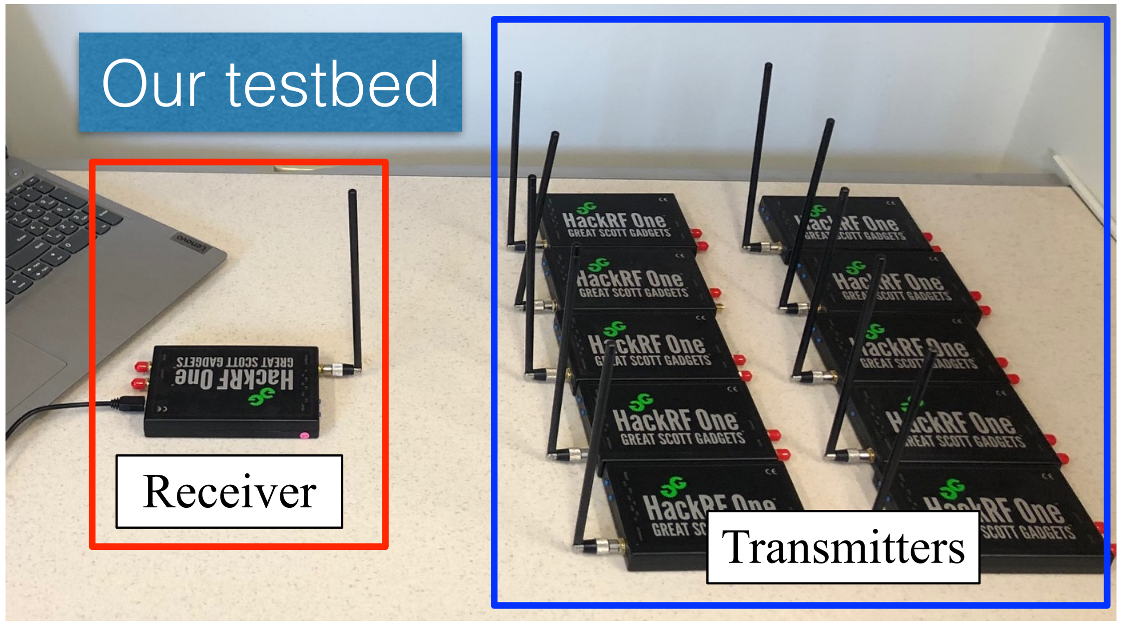
- Source: Day 1; Target: Day 2
- Train ADA with a large amount of RF signals from Day 1 and a small amount of RF signals from Day 2
- Tune k-NN for better classification for Day 2

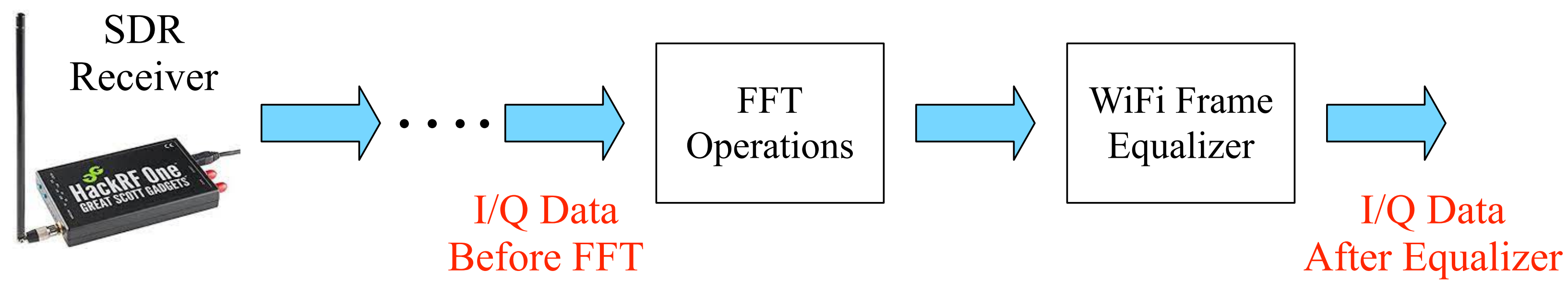


Testbed and Datasets

- **NEU dataset** (from INFOCOM'20):
  - 1 USRP as receiver, 20 USRP as transmitters
  - RF signals from 2 days

- **HackRF-10 dataset** (Ours)
  - 1 receiver, 10 transmitters
  - HackRF One, GNU Radio
  - WiFi, BPSK 1/2, Indoor
  - RF signals from 2 days
  - 3 transmissions per day
  - 30 secs per transmission
  - 3.26 million I/Q samples collected

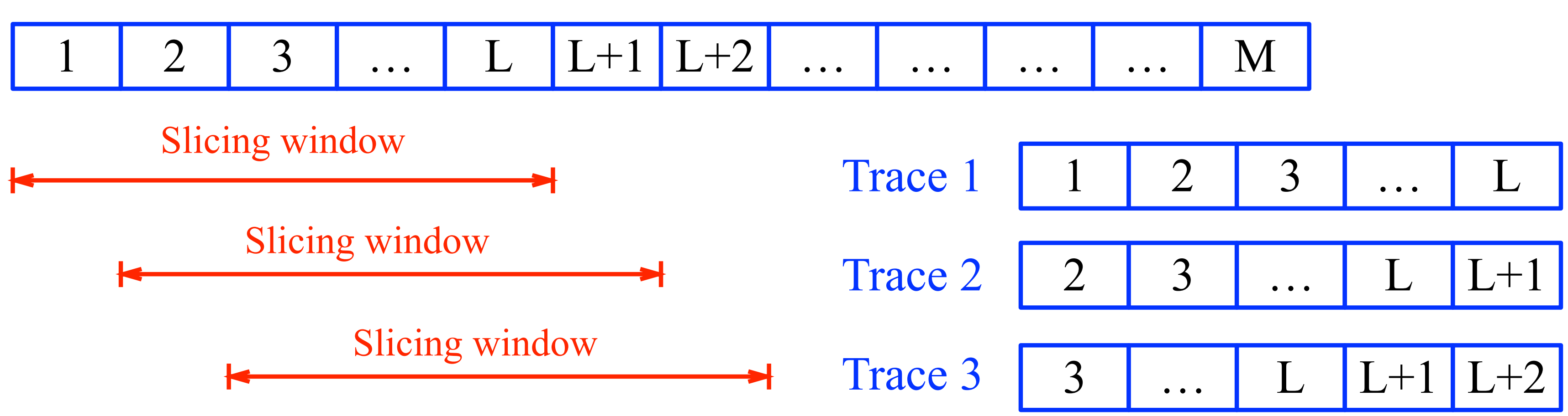




- Collect I/Q samples before FFT and after Equalizer
- Time domain, Frequency domain, Time-Frequency Domain

- Parameters in Pre-Process: Window Size L and Stride s

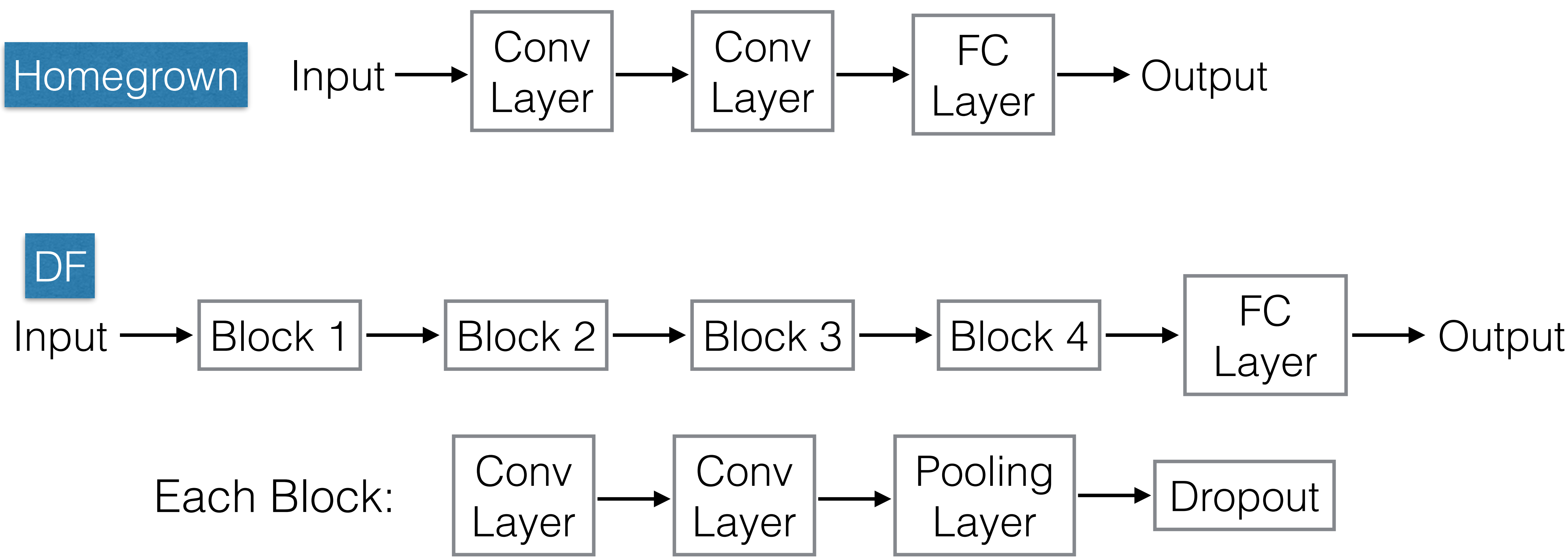
I/Q Samples in One Transmission



Slicing window with size L and stride 1

Evaluation Setting

- Two CNN: Homegrown (INFOCOM'19) and DF (CCS'18)
  - Keras and Tensorflow (Nvidia Titan RTX)
  - Training (64%), Validation (16%), Testing (20%)



Evaluation Metric

- **Accuracy**: Given N test traces, x traces are predicted correctly. Acc = x/N

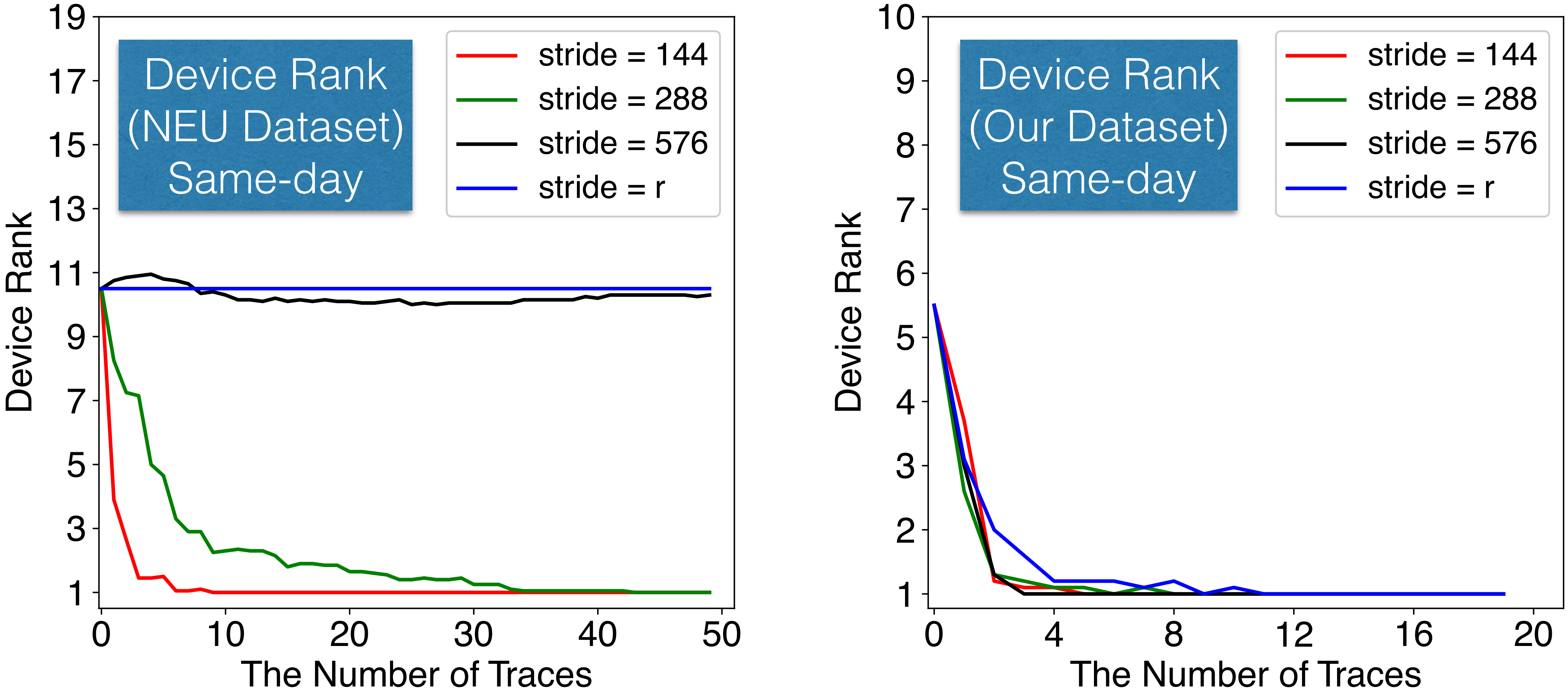
- **Device Rank**: aggregate scores of transmitters over N traces, sort the aggregated scores, report the rank of the correct transmitter

- Why device rank is better than accuracy?
  - Hardware imperfections are difficult to learn
  - Aggregated scores are more robust

The impact of stride s on accuracy
(Time domain, window/trace length L = 288)

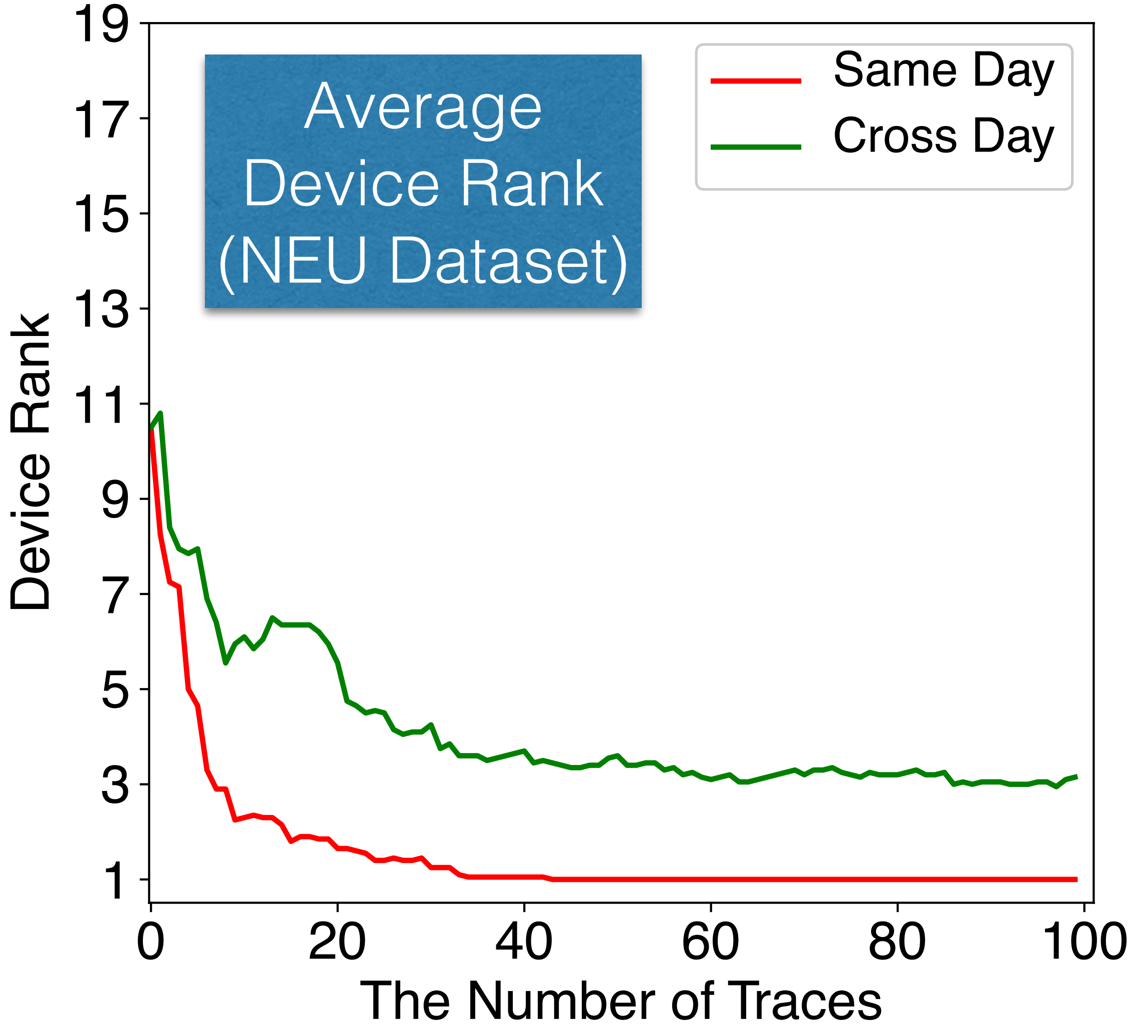
Stride *s*	Neural Networks	NEU dataset (Random guess 5%)		HackRF dataset (Random guess 10%)	
		Same-Day	Cross-Day	Same-Day	Cross-Day
s=1	Homegrown	99.74	6.26	99.76	20.40
	DF	99.95	6.08	99.99	21.85
s=144	Homegrown	26.47	7.59	59.31	24.75
	DF	50.02	6.90	68.63	26.90
s=*L*=288	Homegrown	16.90	8.72	52.31	25.83
	DF	14.24	7.31	60.47	27.80
s=2*L*=596	Homegrown	11.61	8.68	45.93	26.23
	DF	5.88	5.70	47.30	26.86

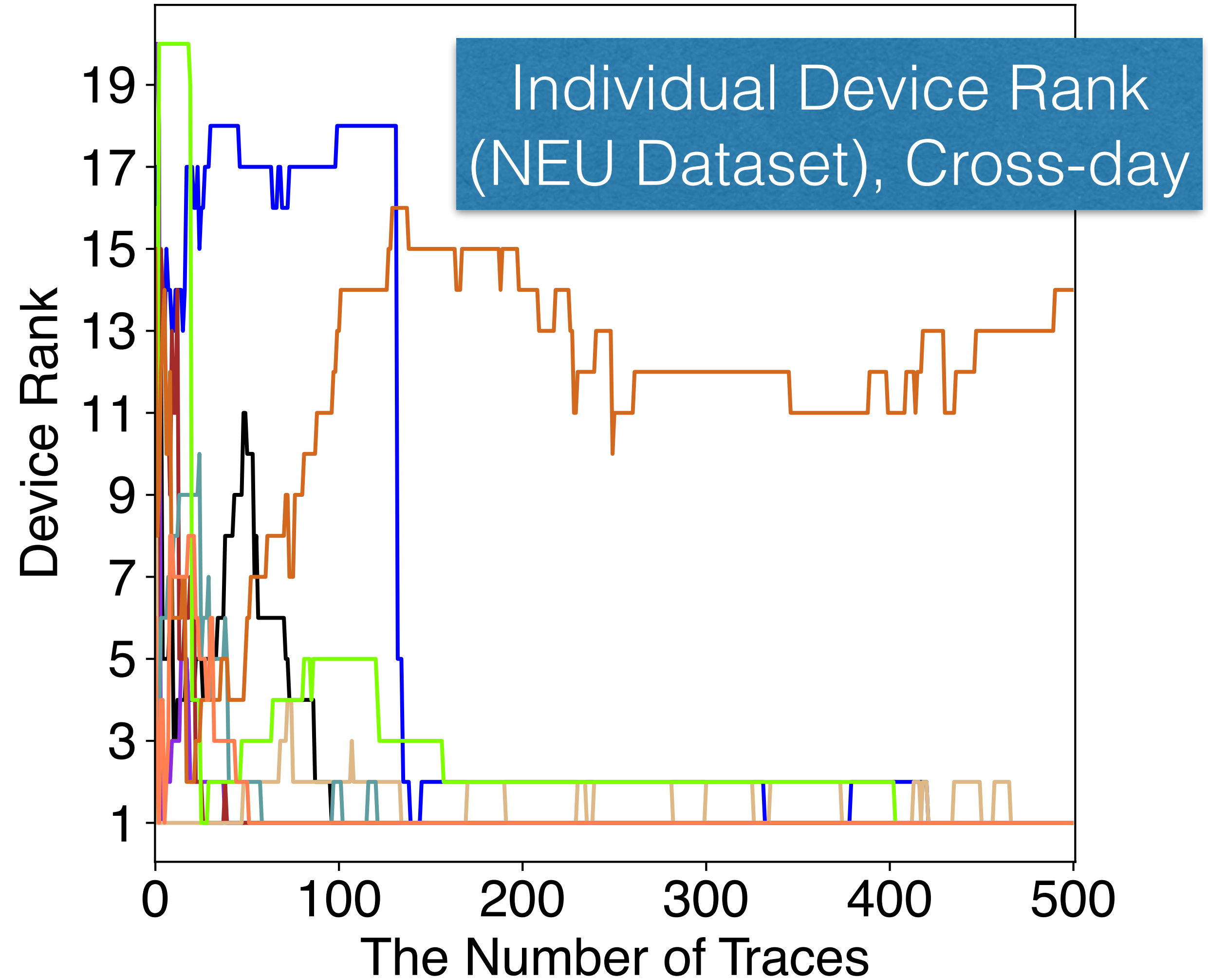
Should choose stride s s.t. there is no overlaps across traces



NEU: stride s = L = 288, accuracy is only 16.9%, device rank still converges to 1 (authenticate correctly) after 35 traces (around 37 milliseconds of RF transmissions)

Low accuracy does not necessarily mean failing to authenticate





Cross-day also affects device rank, but not for every transmitter

The impact of trace length *L* on accuracy
Time domain, stride *s* = *L*

Trace length *L*	Neural Networks	NEU dataset (Random guess 5%)		HackRF dataset (Random guess 10%)	
		Same-Day	Cross-Day	Same-Day	Cross-Day
L=144	Homegrown	22.94	7.13	52.59	25.15
	DF	55.16	7.33	64.43	27.57
L=288	Homegrown	16.96	8.72	52.13	25.83
	DF	14.24	7.31	60.47	27.80
L=576	Homegrown	13.29	9.26	46.27	26.49
	DF	6.56	5.82	57.78	27.90

A greater L should be chosen whenever it is possible

Comparison among different domains of I/Q data

Domain	Neural Networks	HackRF dataset (Random guess 10%)	
		Same-Day	Cross-Day
Time	Homegrown	48.07	13.13
	DF	54.96	14.01
Frequency	Homegrown	50.49	**27.67**
	DF	59.71	**28.74**
Time-Frequency	Homegrown	50.51	12.64
	DF	**60.21**	15.74

Frequency domain is more robust in cross-day

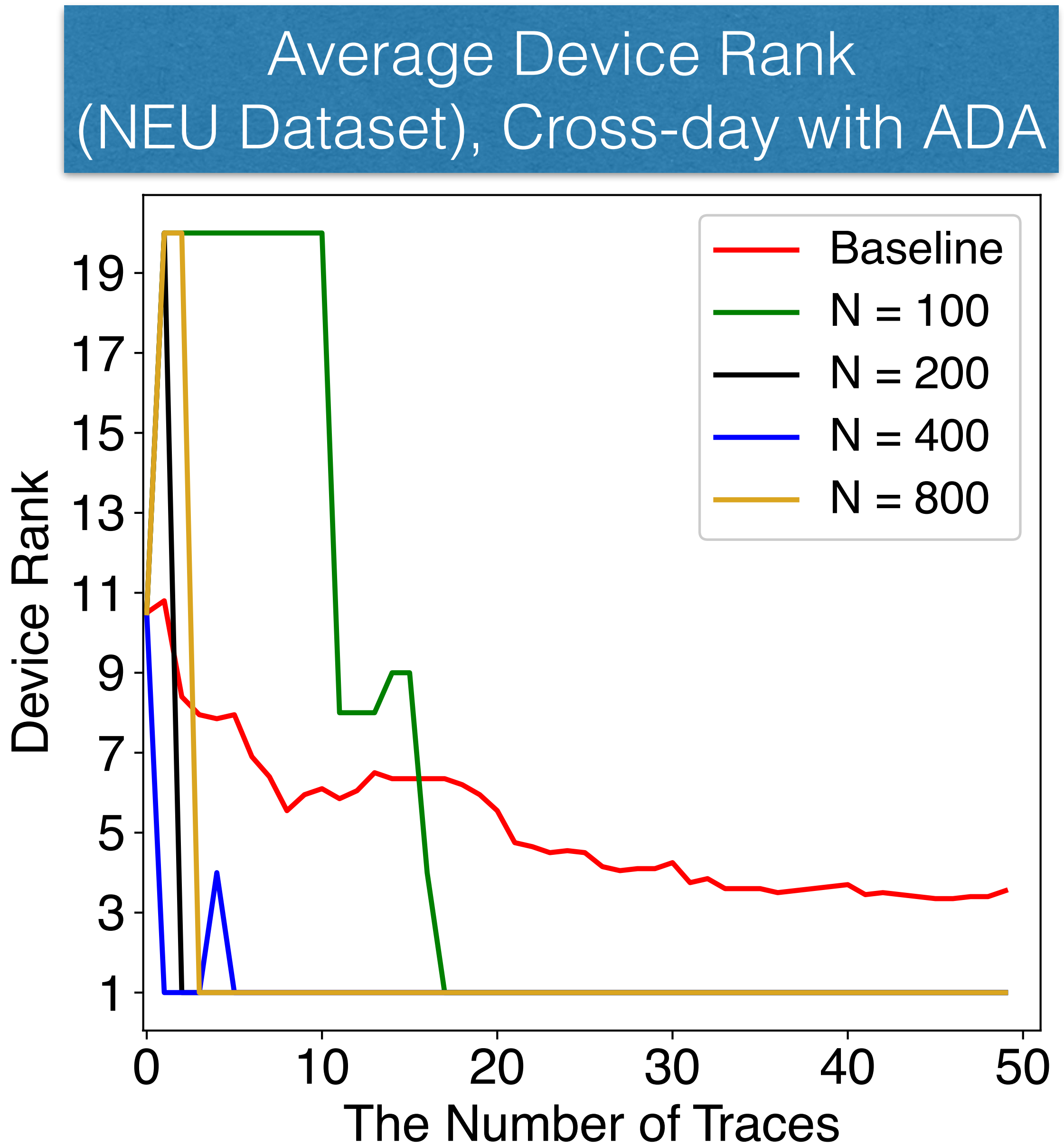
Accuracy in the cross-day (HackRF-10 dataset)

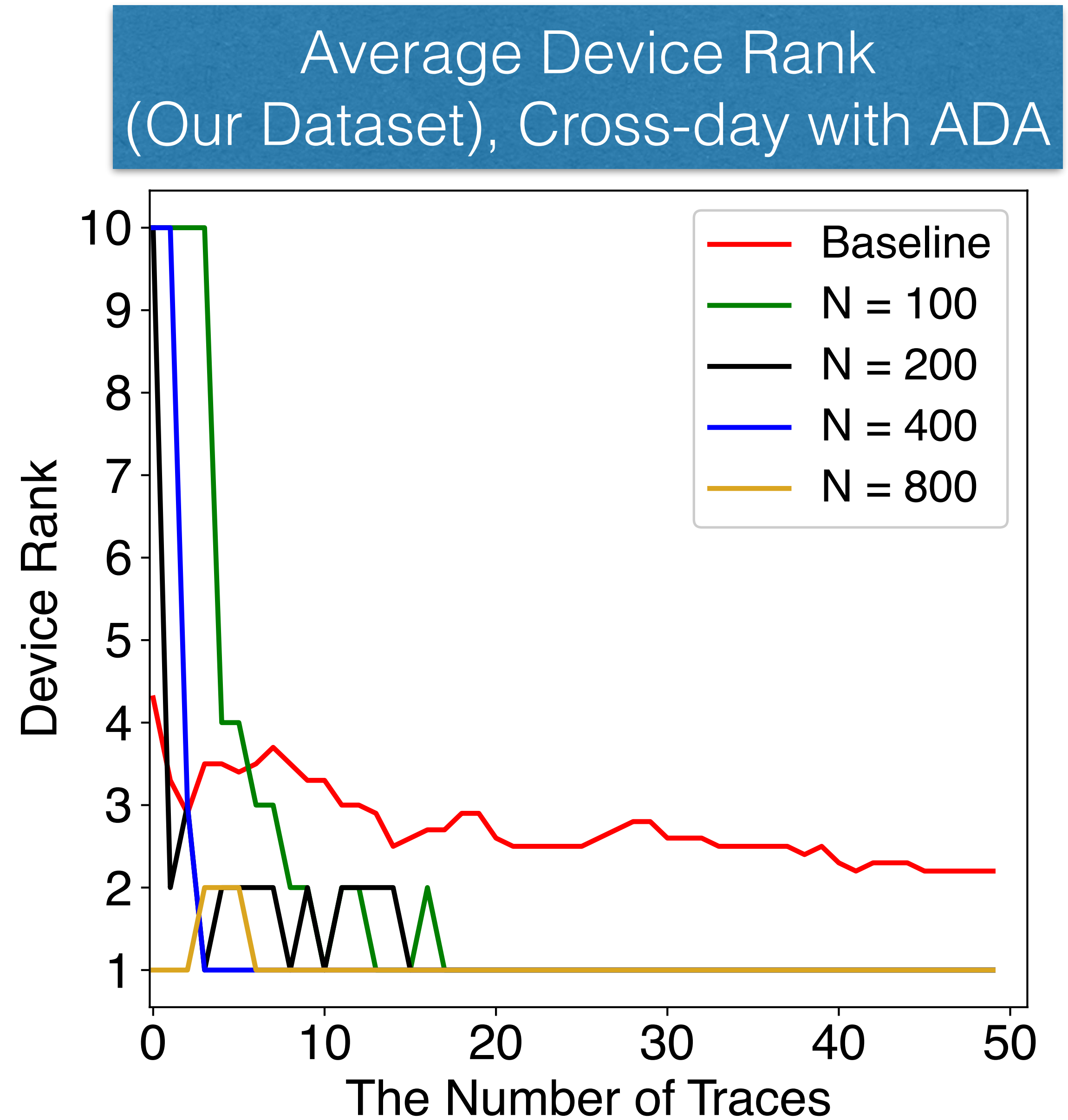
Domain	Method	N					
		0	10	100	200	400	800
Frequency	Fine-tuning	25.98	**34.95**	46.45	49.98	52.56	55.17
	ADA	25.98	33.82	**47.82**	**53.95**	**59.94**	**65.24**
Time-Frequency	Fine-tuning	17.19	**31.46**	**45.60**	50.85	53.62	56.04
	ADA	17.19	29.22	42.42	**54.41**	**58.30**	**64.22**

- Fine-tuning (WiSec'21) v.s. Our method based on ADA
- N: No. of traces per transmitter from Day 2
- Baseline (N = 0): train Day 1 test Day 2 directly

Both methods improve cross-day accuracy

Ours is better when N>=200





- Baseline (N = 0): train Day 1 test Day 2 directly

Device rank in cross-day is also improved by ADA

Discussions and Future Work

- Complex-value neural networks
  - IQ samples are complex values
  - Operations (e.g., max) are not defined over complex values
  - Transfer learning over complex values
- How frequent we need to tune a classifier?
- Can we tune without labeled data from Day 2?

Thank you! Q&A

This work is partially supported by NSF (CNS-1947913)

Code and datasets: https://github.com/UCdasec/RadioNet

University of CINCINNATI

UNIVERSITY OF Nebraska Lincoln