Dec. 2nd, 2020

ITU AI/ML in 5G Challenge webinar series

DIETI. UNIVERSITA' DEGLI STUDI DI NAPOLI FEDERICO II

TOWARD EFFECTIVE NETWORK TRAFFIC ANALYTICS OF MOBILE APPS VIA DEEP LEARNING

Domenico Ciuonzo, Assistant Professor

University of Napoli Federico II, Italy

domenico.ciuonzo@unina.it

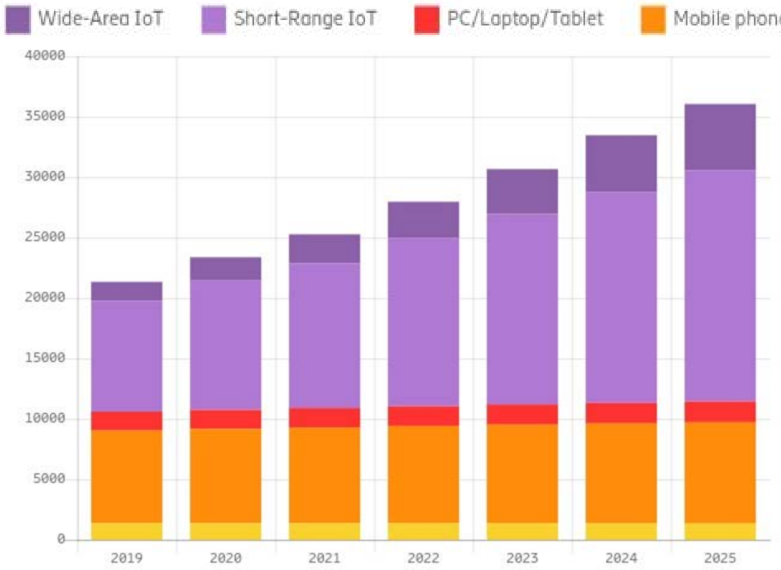
OUTLINE

- Mobile Traffic and Traffic Classification/Prediction (TC/TP)
- Multi-Classification Approaches for Mobile TC
- Mobile TC using Deep Learning (DL)
- The use of Multimodal-DL and Improvements
- Multipurpose TC via Multitask DL
- Reproducibility and Dataset Quality
- Mobile App TP: A first shot
- Take-Home Messages

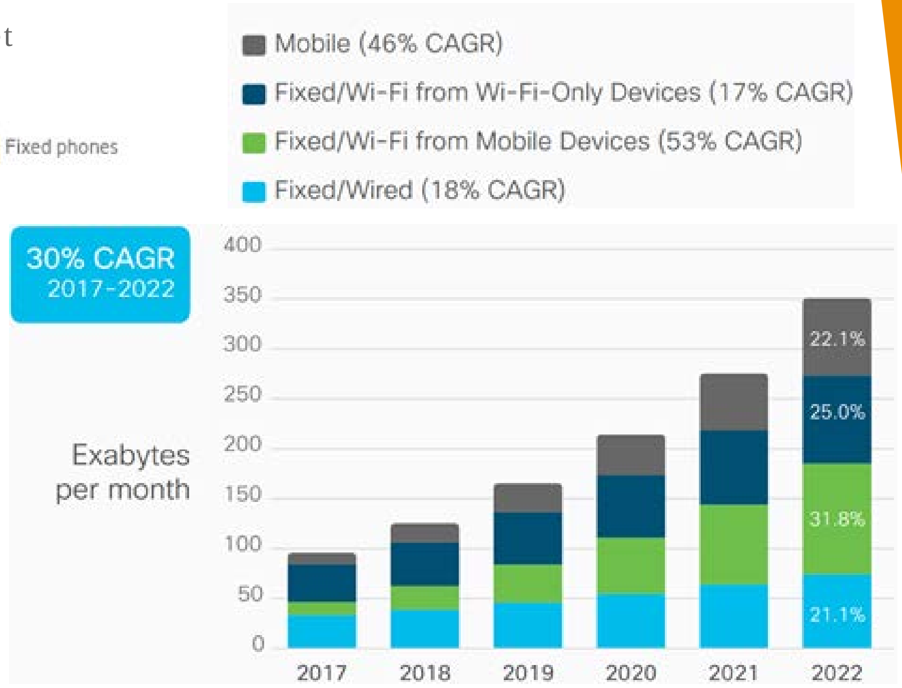
MOBILE TRAFFIC GROWTH

Massive usage of handheld devices has significantly changed the traffic

- traversing home and enterprise networks
- connecting contents and services over the Internet



Source: Ericsson Mobility Report, Jun. 2020

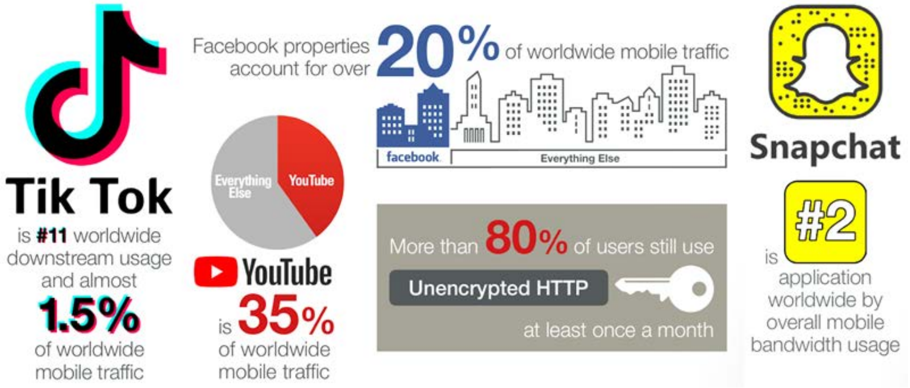


Source: Cisco 2019 VNI Global IP Traffic Forecast 2017-2020

MOBILE TRAFFIC CLASSIFICATION

What is flowing through my (mobile) network?

*Need for **associating** flows (or other classification objects) with the mobile apps that generate them and **predicting** their behaviour*



Source: Sandvine,

The Mobile Internet Phenomena Report, 2019 & 2020

MOBILE TRAFFIC ANALYSIS: MAIN DRIVERS

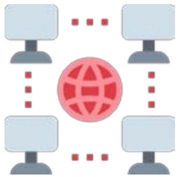
Classification of mobile traffic **provides valuable information** for

- Advertisers
- Insurance companies
- Security agencies
- Infrastructure Operators

But also **raises privacy issues**

- Context-sensitive apps
- Bring your own device policy
- Indiscriminate surveillance

Profiling



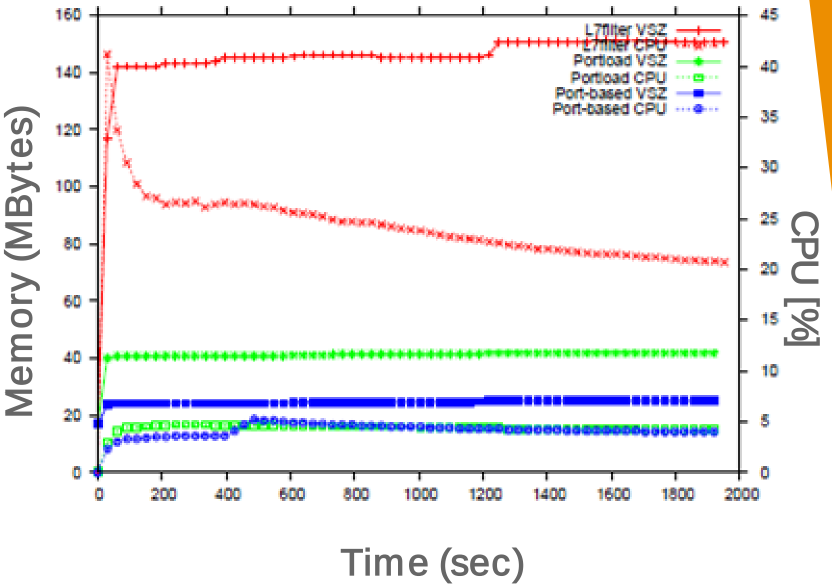
Monitoring & Mgt

EARLY DAYS OF TC: PORT+DPI

- PortLoad* (fast & privacy-friendly):
  - needs the 1st packet only (with direction)
  - uses fixed fields (protocol)
  - uses few data (fixed values in fixed positions, such as port inspection)

Classifier	Accuracy on applications sessions	bytes
PortLoad	74.24%	97.83%
Port-based	19.57%	25.12%

*Patent No.: NA2010AOOOO11



Classifier	Mean Time (μsec)	Mean Time (*vs* Port-based)	Variance (μsec^2)
Port-based	2.48	1.0	0.88
PortLoad	6.99	2.8	11.15
L7-Filter	211.4	85.2	47057.88

MOBILE TRAFFIC ANALYSIS:
MAIN CHALLENGES

Many apps

Heterogeneous & Dynamic

Background &
3rd-party services

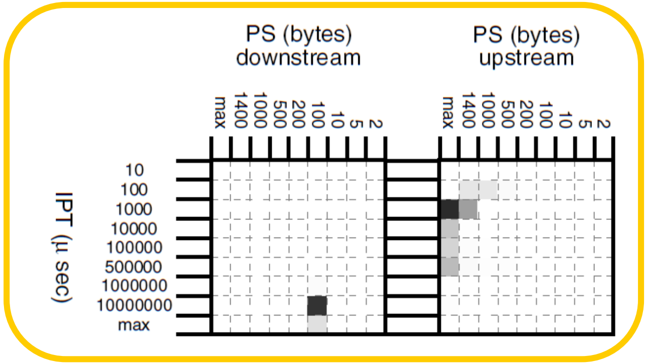
Encryption
(TLS)

TRAFFIC CLASSIFICATION: FEATURE DESIGN

Statistical features

Feature Extractor

- PL-IAT sequences
- PL-IAT histograms
- PL-IAT transition probabilities
- Other features (packet ratio, etc.)

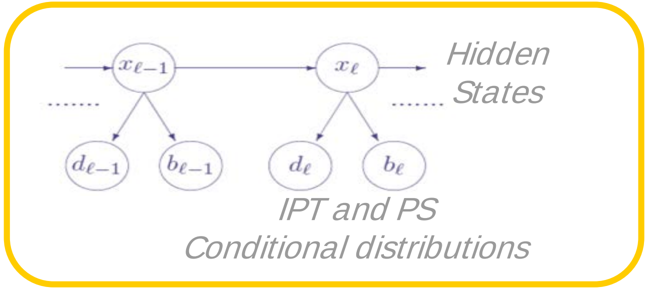


ML Algorithm

Machine Learning

Classifiers

- k-NN / K-dimensional trees
- SVM
- Bayesian Approaches

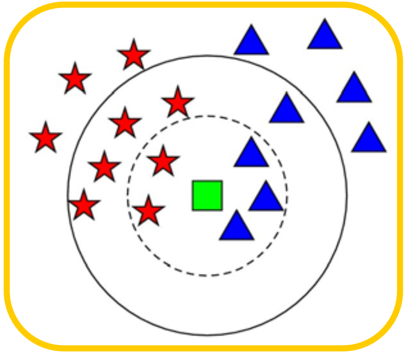


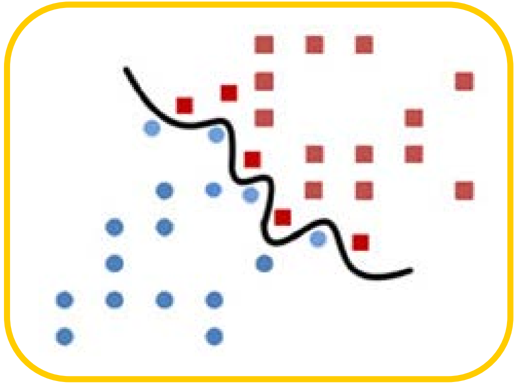
TRAFFIC CLASSIFICATION: FEATURE DESIGN

Statistical features

Feature Extractor

- PL-IAT sequences
- PL-IAT histograms
- PL-IAT transition probabilities
- Other features (packet ratio, etc.)

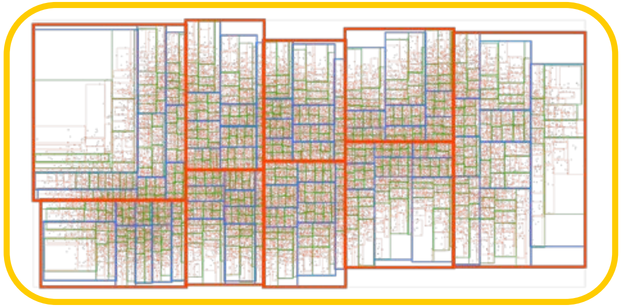




ML Algorithm

Machine Learning Classifiers

- k-NN / K-dimensional trees
- SVM
- Bayesian Approaches



TAKING THE BEST FROM EACH STATE-OF-THE-ART CLASSIFIER

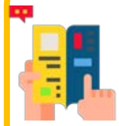
Objective

Improve

Mobile TC

Performance

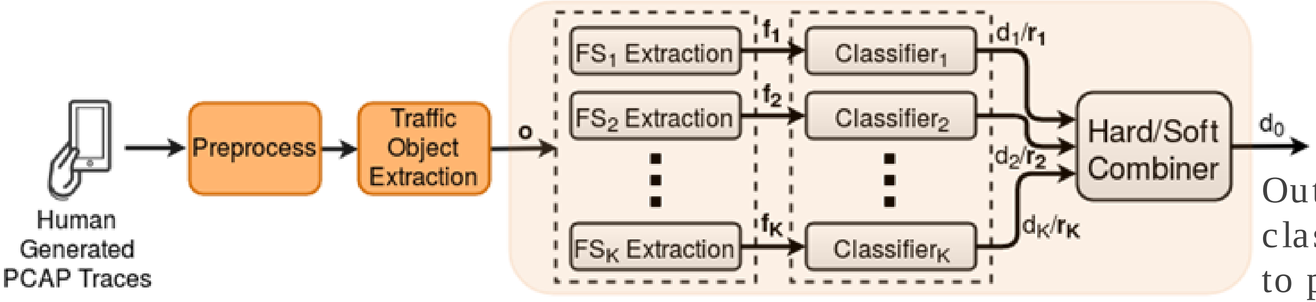
How?



Select promising classification algorithms and related features from state-of-the-art

Design a MCS that takes the best classifiers and adopt state-of-the-art combining techniques

Multi-Classification System (MCS)



Outputs from Machine Learning classifiers can be combined to perform Multi-Classification

DATASET DESCRIPTION

DL classifiers are compared on three datasets of **real-user traffic** and labeling each trace with the generating app **run separately**

1 Android

- Multi-class
- Collected by a mobile solutions provider
- Apr. 2015 – Jan. 2017
- 49 apps
- 77.5k biflows

2 iOS

- Multi-class
- Collected by a mobile solutions provider
- Jul. 2014 – Jan. 2017
- 45 apps
- 40.5k biflows

3 FB/FBM

- Binary
- Collected @ ARCLAB FII
- > 100 users
- May. 2017 – Mar. 2018
- Facebook (FB) & FB Messenger (FBM)
- 17.0k FB biflows (62%)
- 10.5k FBM biflows (38%)

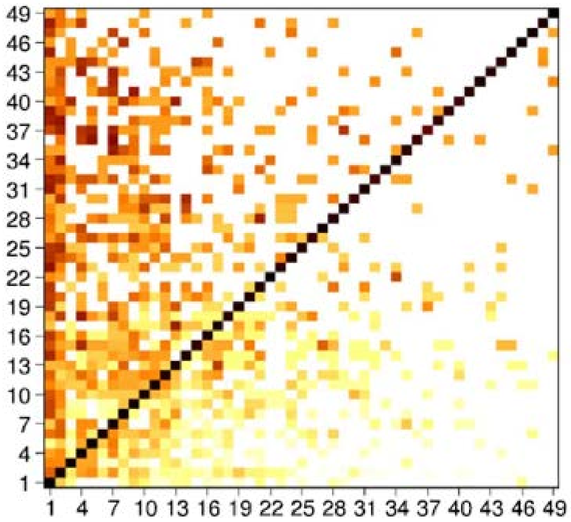
MCS TRAFFIC CLASSIFICATION: PERFORMANCE

49 apps	Accuracy [%]	Precision [%]	Recall [%]	F-Measure [%]
Oracle	87.6	N/ D	83.6	N/ D
Best Soft Combiner (KL-weights)	79.2	80.6	73.6	83.7
Best Hard Combiner (Naive Bayes)	75.0	77.4	69.7	75.7
Best Classifier (Random Forest)	72.8	74.7	64.1	72.3

MCS TRAFFIC CLASSIFICATION: PERFORMANCE

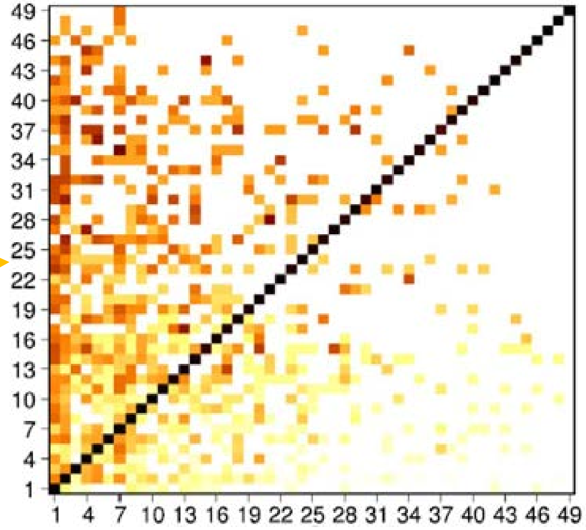
Best base classifier: *Tay_RF*

- Accuracy: 72.8
- F-measure: 72.3



Best combiner: *KL-Weights*

- Accuracy: 79.2
- F-measure: 83.7



BEYOND MACHINE LEARNING (ML) TRAFFIC CLASSIFIERS

Input → Feature Extractor → Features → ML Algorithm → Output

Traditional ML flow

Standard ML

relies on domain-expert driven handcrafted features

- ➔ Time-consuming process
- ➔ Unsuited to automation
- ➔ Rapidly outdated

Difficulty to design **accurate** and **up-to-date** mobile traffic classifier

Input → Deep Learning Algorithm → Output

Deep Learning flow

Deep Learning (DL)

allows to train classifiers directly from input data

- ➔ Automatic hierarchical feature extraction
- ➔ Reduced preprocessing effort

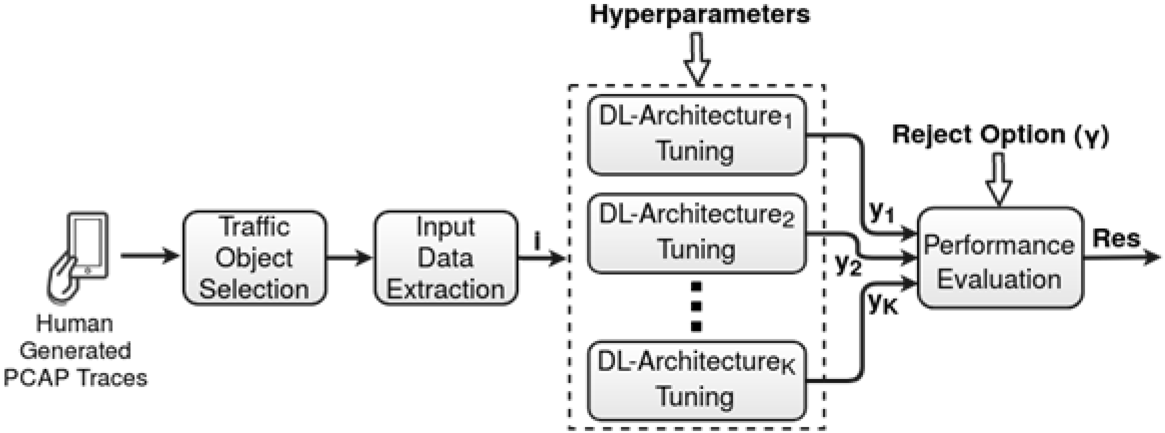
Stepping stone toward the achievement of **high performance** in mobile TC

MOBILE TC USING DL:
RESEARCH GOAL

Naïve adoption of DL techniques to mobile TC may imply **misleading** design choices and lead to **biased** conclusions

We propose the **design** of DL-based mobile traffic classifiers resorting on a **systematic framework** expressly developed for their comparison

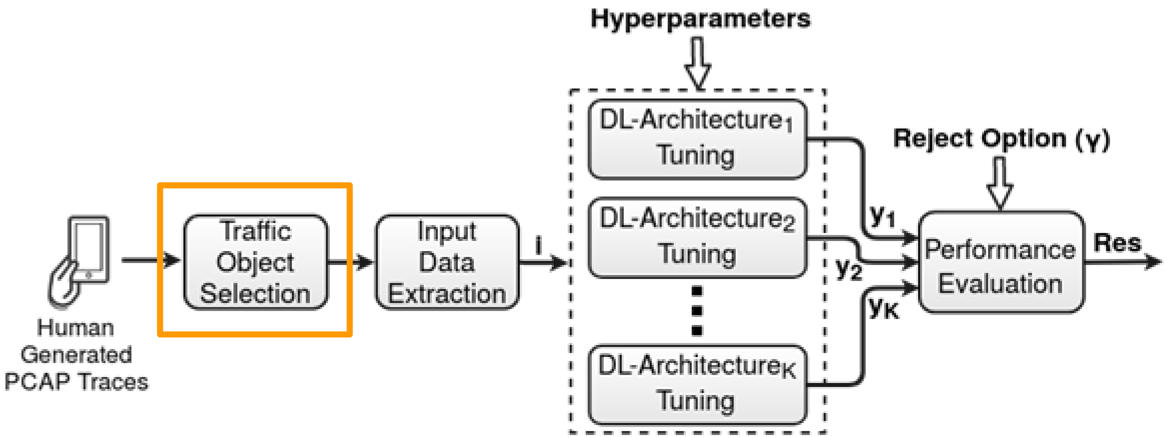
DEFINING DL-BASED TC WORKFLOW



The proposed framework dissects the TC problem from different viewpoints

- TC object adopted
- Type and amount of input data fed to the DL classifier
- DL architecture employed
- Required set of performance measures

WHICH TRAFFIC OBJECT?

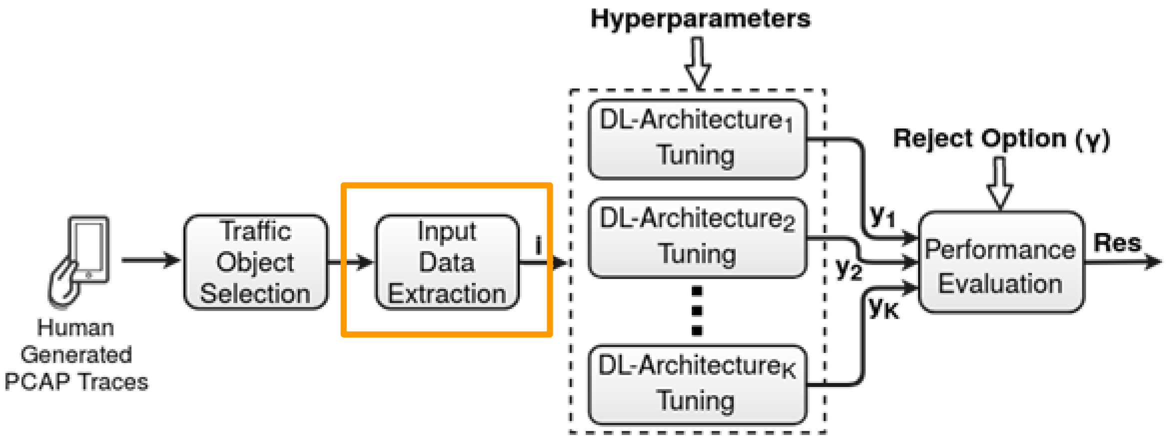


The definition of a specific TC object determines how the traffic is segmented into multiple discrete traffic units

The majority of works approaching TC using DL considered

- Flows
- **Biflows**

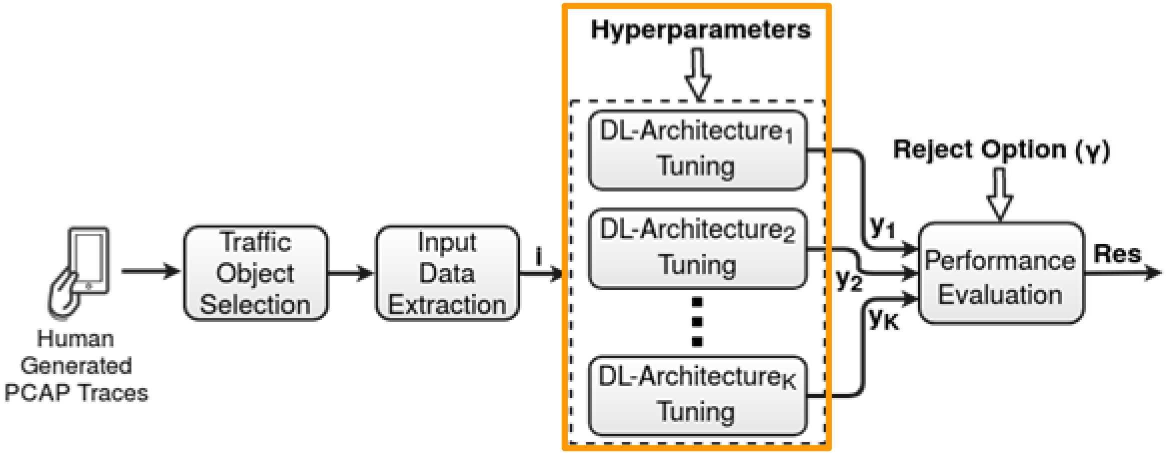
WHICH & HOW MUCH INPUT DATA?



There is **no feature extraction**, only need to provide the input

- First N bytes of TC-object payload $[N \times 1] \rightarrow$ **L7-N**
  First 784/1000 bytes of L7 payload of each biflow
- First N bytes of TC-object raw data $[N \times 1] \rightarrow$ **ALL-N**
  First 784/1000 bytes of PCAP raw data of each biflow
- Informative fields of first N_p packets $[20 \times 6] \rightarrow$ **MAT**
  (1) Source port, (2) Destination port, (3) Payload length,
  (4) TCP window size, (5) Inter-arrival time, (6) Packet direction

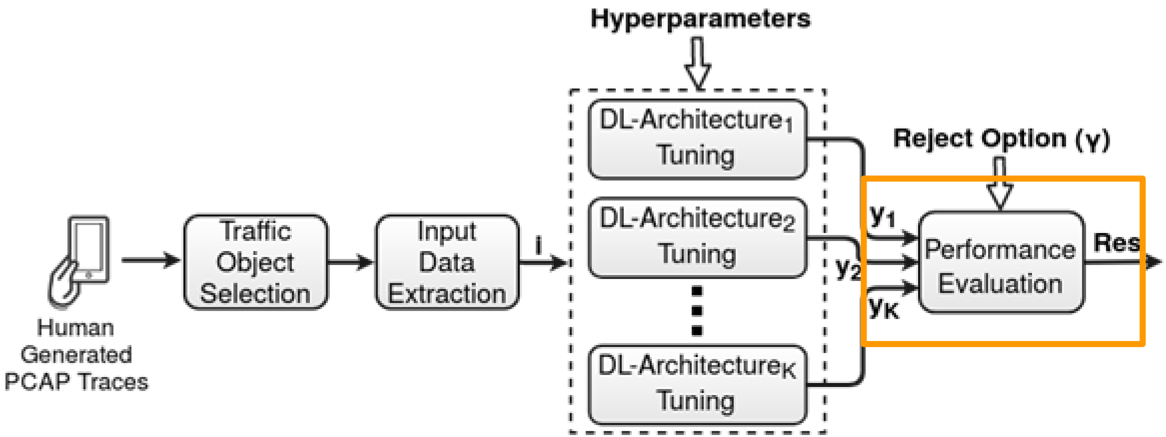
WHICH DL ARCHITECTURE?



DL classifiers are trained to minimize categorical cross-entropy

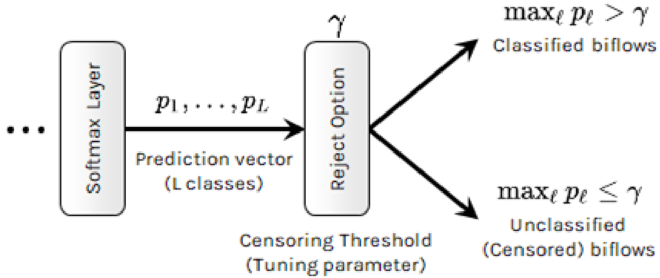
- Stacked AutoEncoder (SAE) fed with L7-1000 [1]
- Convolutional Neural Network (CNN)
  - 1D-CNN fed with L7-784 and ALL-784 [2]
  - 2D-CNN fed with L7-784, ALL-784 [3], and MAT [4]
- Long Short-Term Memory (LSTM) fed with MAT [4]
- Hybrid DL architecture (LSTM + 2D-CNN) fed with MAT [4]

HOW TO EVALUATE PERFORMANCE?

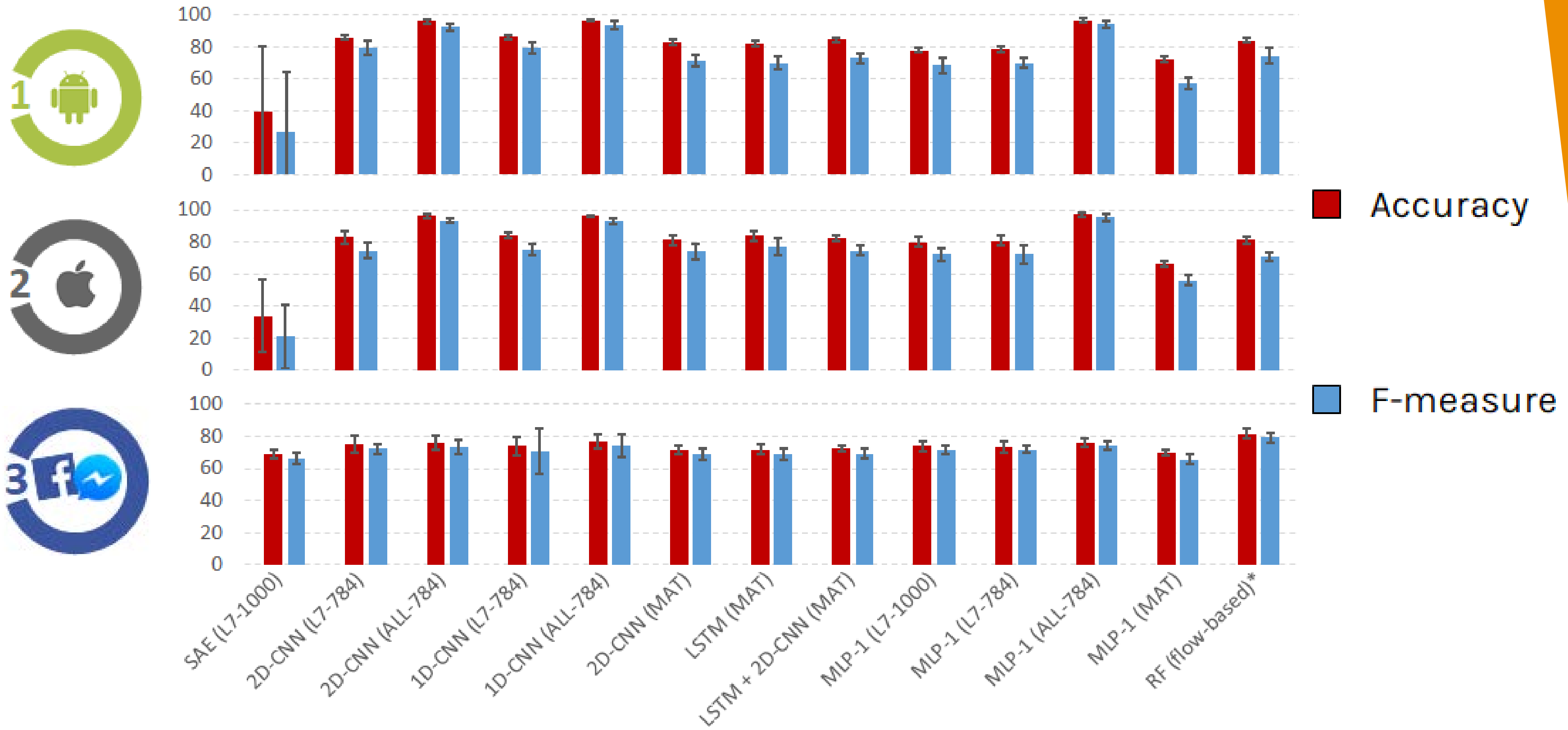


Comparison of DL classifiers for mobile TC benefits from a
comprehensive performance evaluation framework
based on a *stratified 10-fold validation*

- Accuracy
- Macro F-measure
- Top-K accuracy
- Confusion Matrix

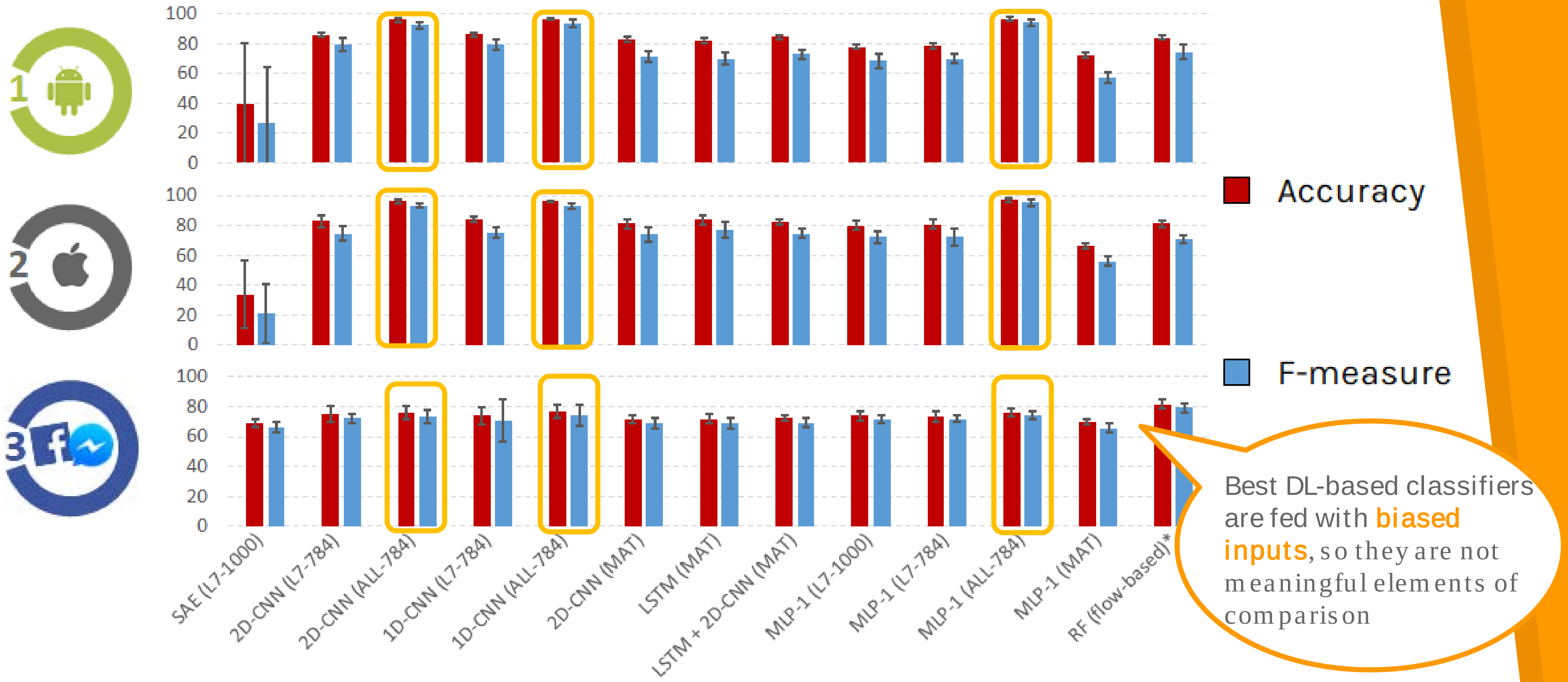


THE BIGGER PICTURE ON PERFORMANCE



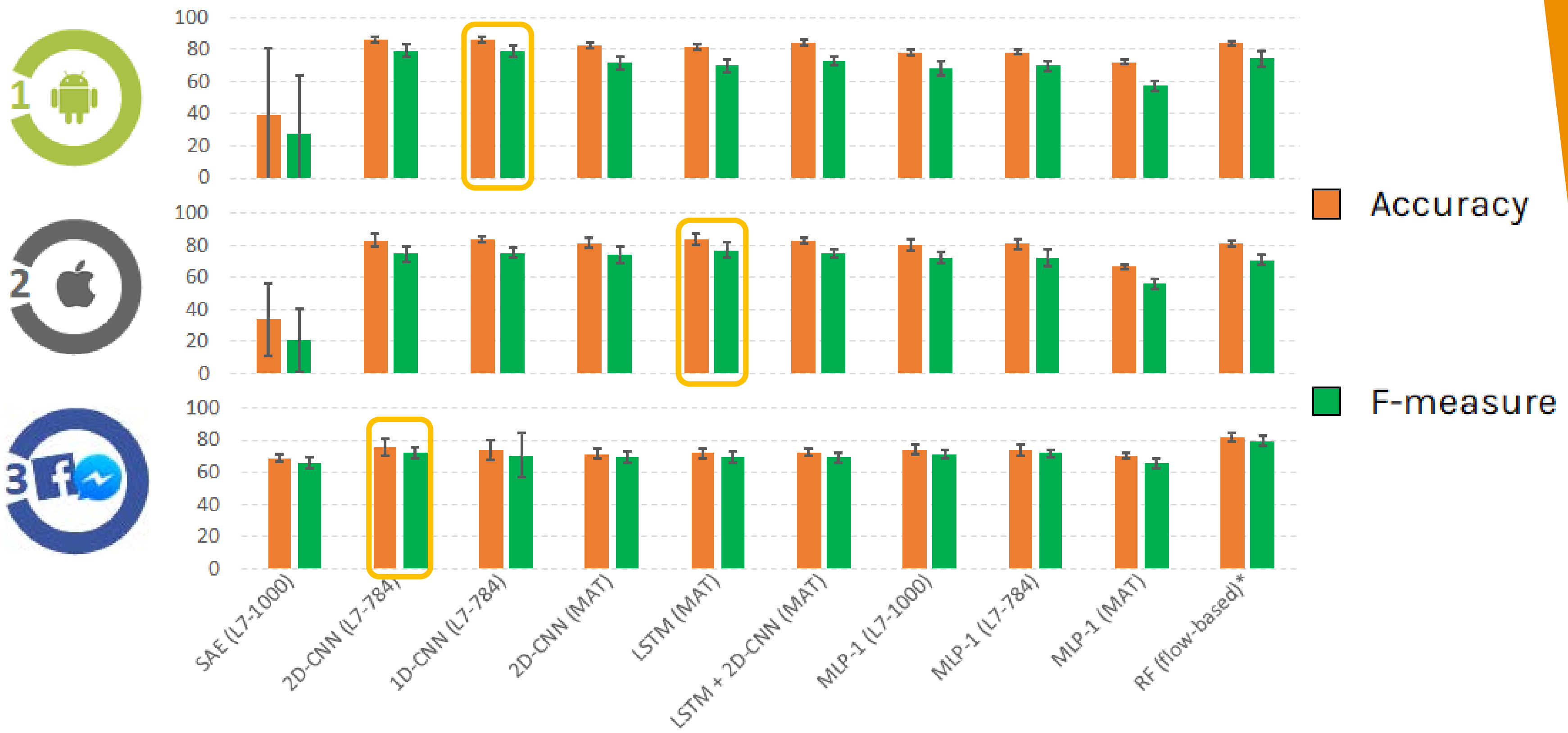
*Taylor et al., "Appscanner: Automatic fingerprinting of smartphone apps from encrypted network traffic"

DON’T TRUST EVERY INPUT DATA



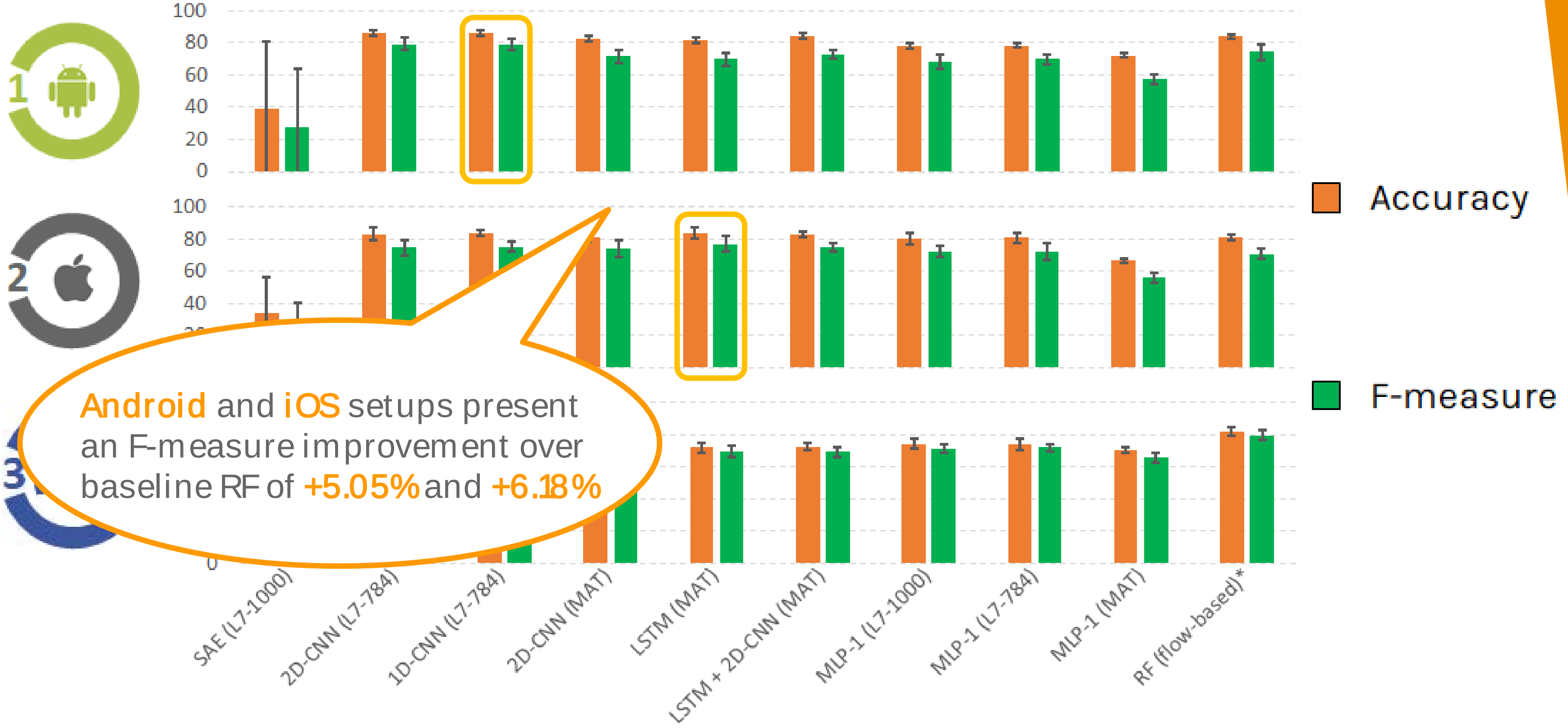
*Taylor et al., “Appscanner: Automatic fingerprinting of smartphone apps from encrypted network traffic”

CAN PERFORMANCE BE IMPROVED W.R.T. BASELINE CLASSIFIER?



*Taylor et al., "Appscanner: Automatic fingerprinting of smartphone apps from encrypted network traffic"

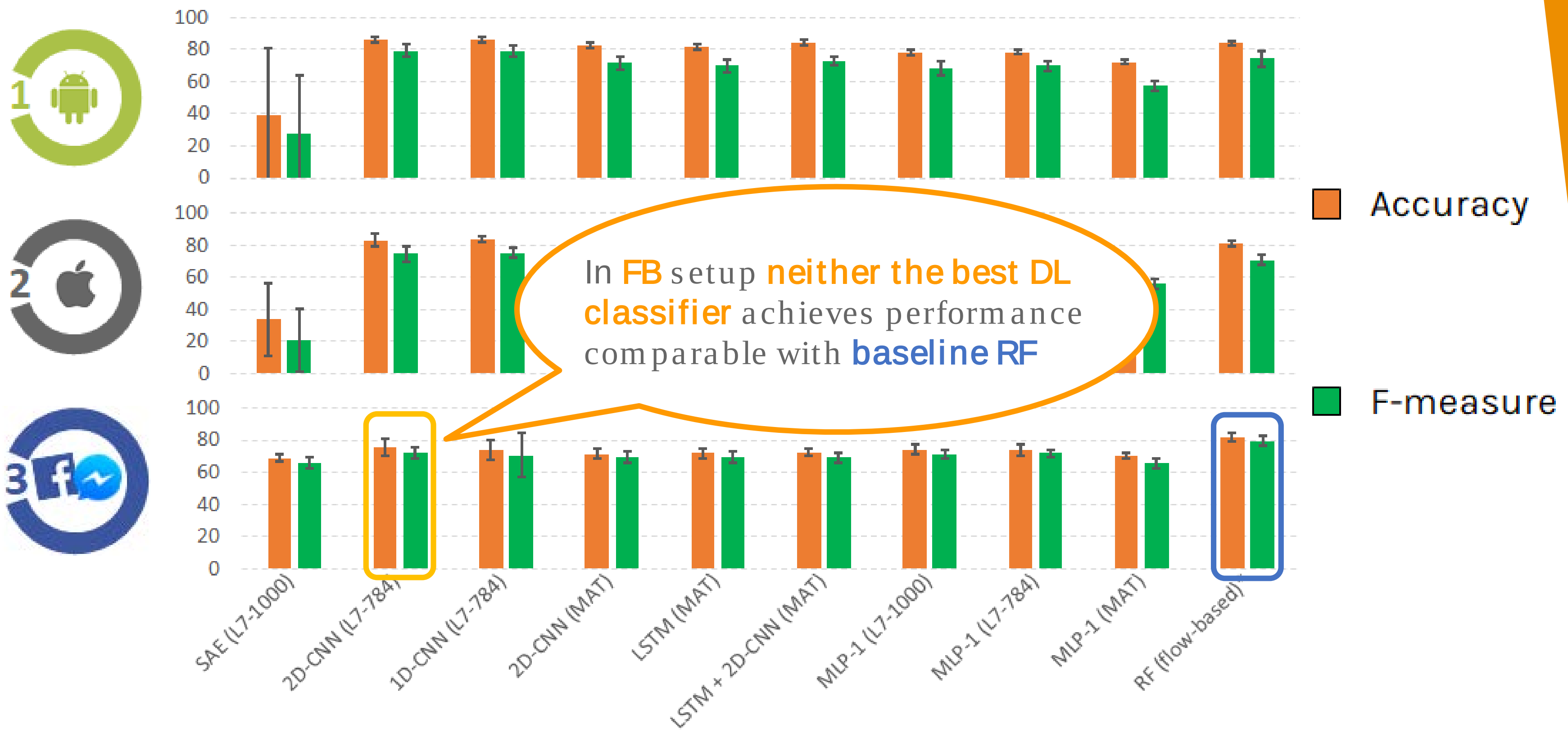
CAN PERFORMANCE BE IMPROVED W.R.T. BASELINE CLASSIFIER?



Android and iOS setups present an F-measure improvement over baseline RF of +5.05% and +6.18%

*Taylor et al., "Appscanner: Automatic fingerprinting of smartphone apps from encrypted network traffic"

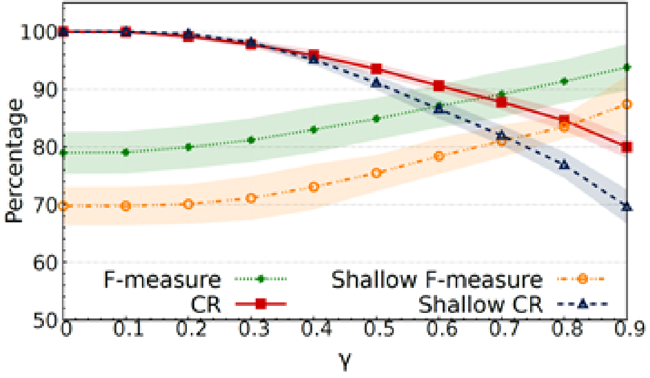
CAN PERFORMANCE BE IMPROVED W.R.T. BASELINE CLASSIFIER?



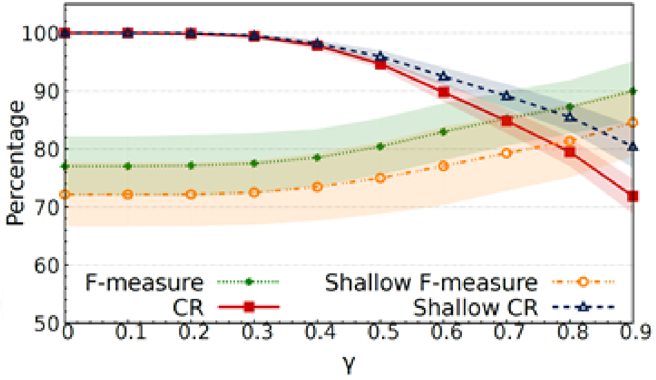
*Taylor et al., "Appscanner: Automatic fingerprinting of smartphone apps from encrypted network traffic"

NO NEED TO CLASSIFY ALL THE INSTANCES: REJECT OPTION

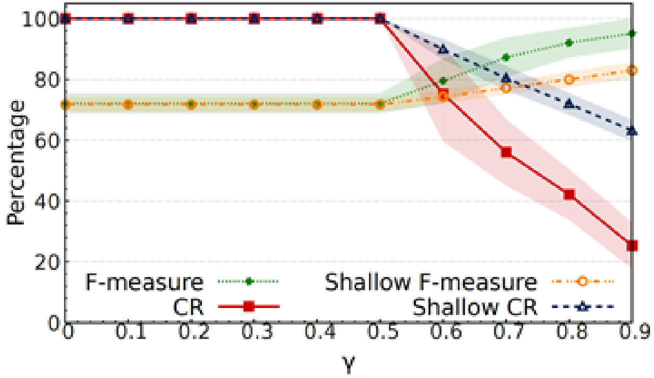
1D-CNN (L7-784)



LSTM (MAT)



2D-CNN (L7-784)



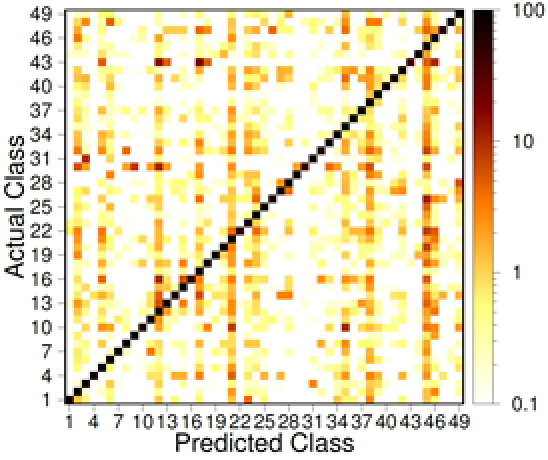
Performance improvement with a negligible ratio of unclassified samples evident only for multi-class datasets

To achieve **>84% F-measure**, rejected

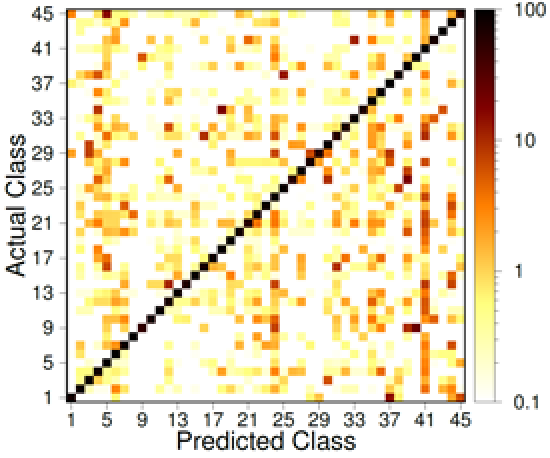
- **10%** of flows for Android and iOS
- **40%** of flows for FB/FBM

GOING DEEP: CONFUSION MATRICES

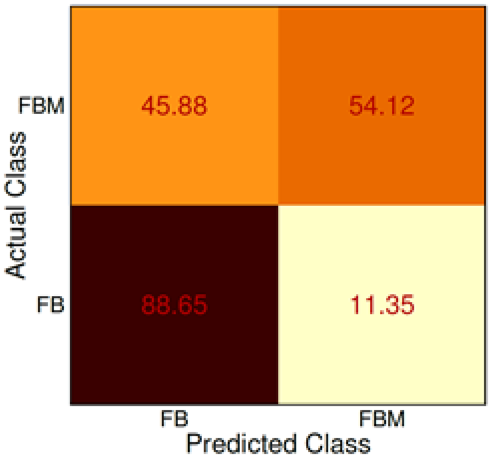
1 1D-CNN (L7-784)



2 LSTM (MAT)



3 2D-CNN (L7-784)



1D-CNN (L7-784) and LSTM achieve almost-uniform error patterns

2D-CNN (L7-784) entails a prediction imbalance toward FB app as a consequence of the higher number of samples in the dataset

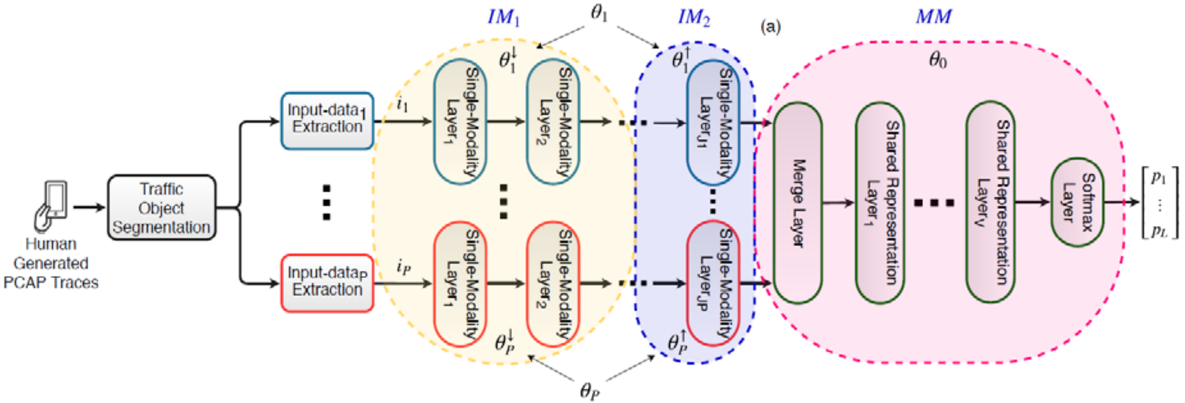
SOME THOUGHTS

- Existing proposals only exploited one kind of traffic “modality”
- Many of the architectures proposed were ad-hoc
- In some cases, the class imbalance effect is strong

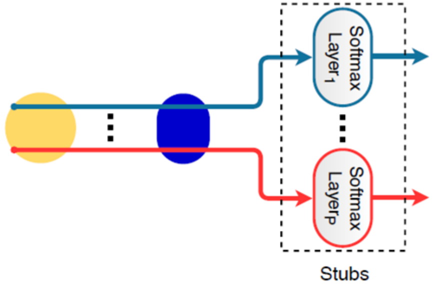
WHAT IS NEXT? MIMETIC

MultI-modal DL-based MobilE TraffIc Classification

- Capitalization of heterogeneous of traffic data
- Capturing both **intra-** and **inter-modalities**



(I) Pre-training



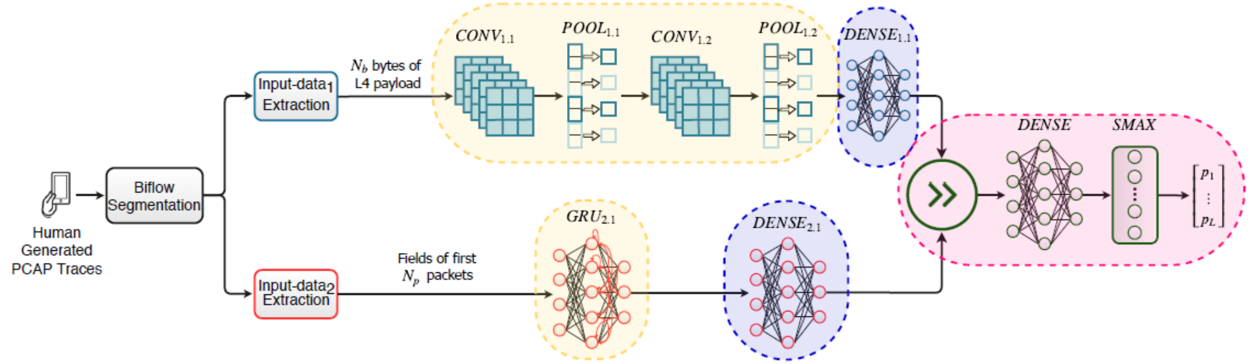
(II) Fine-tuning

- Architectural Overview

WHAT IS NEXT? MIMETIC

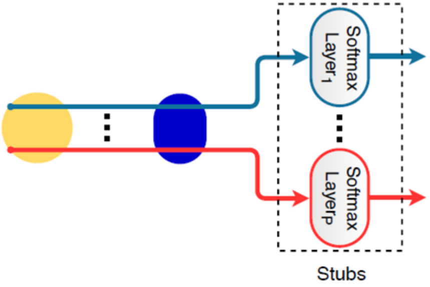
MultI-modal DL-based MobilE TraffIc Classification

- Capitalization of heterogeneous of traffic data
- Capturing both intra- and inter-modalities

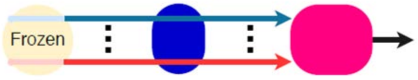


- MIMETIC Instance

(I) Pre-training



(II) Fine-tuning



* With cost-sensitive learning!

MIMETIC PERFORMANCE

FB/FBM

+116% G-Mean

	Architecture	Accuracy	F-measure	G-mean
	MIMETIC	**79.98 (± 0.49)**	**79.63 (± 0.51)**	**79.53 (± 0.60)**
I	1D-CNN [99] (L7-784)	76.37 (± 0.73)	75.56 (± 1.01)	74.79 (± 1.76)
I	HYBRID [96] (MAT-20)	74.26 (± 0.98)	73.23 (± 0.95)	72.18 (± 1.05)
II	MLP-1 (L7-784)	74.46 (± 0.88)	73.89 (± 0.86)	73.55 (± 0.89)
II	MLP-1 (MAT-20)	68.93 (± 1.32)	67.86 (± 0.94)	66.98 (± 0.75)
III	Tay_RF [42] (flow-based)	79.56 (± 0.62) ♦	78.73 (± 0.62) ♦	78.37 (± 0.76) ♦
IV	MV	75.13 (± 0.92)	74.48 (± 1.14)	74.02 (± 1.65)
IV	SOA	78.86 (± 0.79) ‡	78.37 (± 1.00) ‡	78.06 (± 1.61) ‡
IV	TLF	74.61 (± 1.57)	73.60 (± 1.80)	72.59 (± 2.14)
	MIOB-C	+ 0.42 (± 0.65)	+ 0.90 (± 0.68)	+ 1.16 (± 0.99)
	MIOB-FT	+ 1.12 (± 0.89)	+ 1.26 (± 1.14)	+ 1.47 (± 1.84)

(I) Best single-modality

(II) Shallow NN

(III) ML state-of-the-art

(IV) Classifier fusion

(MIOB-C)

Max Gain over best Classifier

(MIOB-FT)

Max Gain over best fusion technique

MIMETIC PERFORMANCE

2 Android

3 iOS

+8.66% F-measure on the iOS dataset

	Architecture	Android			iOS		
		Accuracy	*F-measure*	*G-Mean*	*Accuracy*	*F-measure*	*G-Mean*
	MIMETIC	**89.49 (± 0.32)**	**81.51 (± 0.93)**	**91.96 (± 0.95)**	**89.14 (± 0.82)**	**82.99 (± 1.14)**	**92.25 (± 0.84)**
I	1D-CNN [99] (L7-784)	85.70 (± 0.45)♦	78.68 (± 1.20)♦	86.82 (± 0.87)♦	82.64 (± 1.63)♦	74.34 (± 1.29)♦	84.00 (± 1.31)♦
I	HYBRID [96] (MAT-20)	77.95 (± 0.41)	64.52 (± 1.17)	76.35 (± 1.45)	69.17 (± 0.64)	58.75 (± 0.76)	72.17 (± 0.75)
II	MLP-1 (L7-784)	78.71 (± 0.65)	69.79 (± 1.17)	81.52 (± 1.38)	77.16 (± 0.63)	67.61 (± 1.07)	80.11 (± 0.99)
II	MLP-1 (MAT-20)	64.94 (± 0.47)	48.26 (± 0.96)	63.10 (± 1.07)	54.42 (± 0.63)	40.86 (± 1.04)	57.56 (± 1.03)
III	Tay_RF [42] (flow-based)	84.78 (± 0.30)	75.49 (± 0.89)	83.86 (± 0.58)	80.77 (± 0.84)	72.39 (± 1.39)	81.88 (± 1.27)
IV	MV	80.41 (± 0.40)	71.28 (± 0.85)	81.74 (± 0.77)	77.24 (± 0.62)	66.49 (± 0.97)	78.92 (± 0.97)
IV	SOA	87.08 (± 0.29)‡	80.07 (± 0.81)‡	87.00 (± 0.80)‡	84.68 (± 0.55)‡	75.94 (± 1.10)‡	84.15 (± 0.96)‡
IV	TLF	68.87 (± 1.05)	48.82 (± 1.92)	62.55 (± 1.86)	62.01 (± 0.97)	39.07 (± 1.52)	54.07 (± 1.94)
	MIOB-C	+ 3.79 (± 0.59)	+ 2.83 (± 1.66)	+ 5.14 (± 1.06)	+ 6.50 (± 2.12)	+ 8.66 (± 1.77)	+ 8.25 (± 1.72)
	MIOB-FT	+ 2.40 (± 0.48)	+ 1.44 (± 1.56)	+ 4.96 (± 1.46)	+ 4.46 (± 1.01)	+ 7.05 (± 1.43)	+ 8.10 (± 1.27)

(MIOB-C)

Max Gain over best Classifier

(MIOB-FT)

Max Gain over best fusion technique

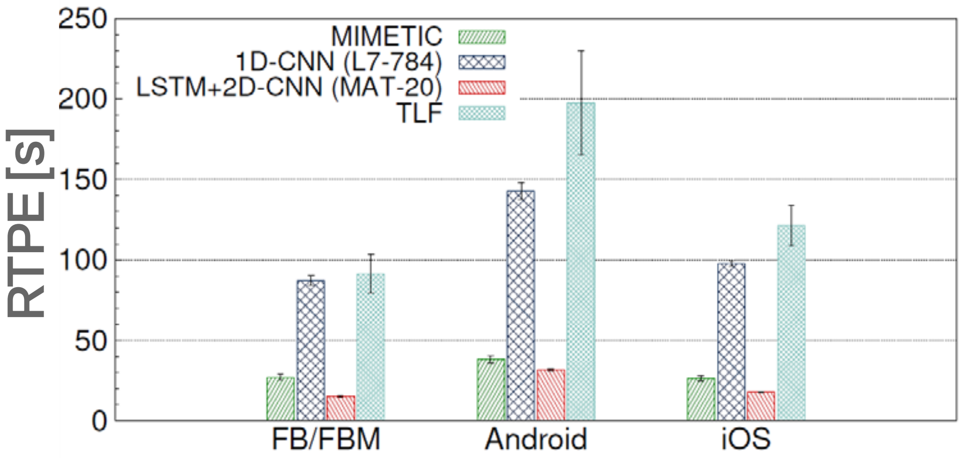
(I) Best single-modality

(III) ML state-of-the-art

(II) Shallow NN

(IV) Classifier fusion

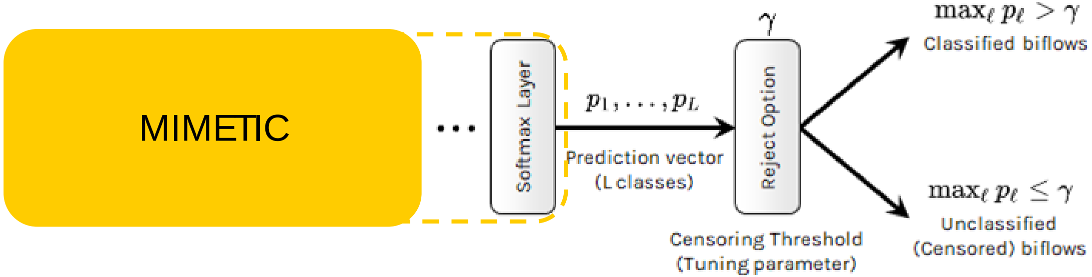
MULTIMODAL-DL HAS
LOWER TRAINING COMPLEXITY



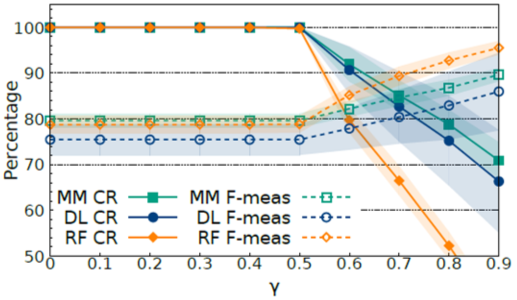
No. of parameters **[Mi]**	FB/ FBM	Android	iOS
MIMETIC	0.93	162	161
Best DL (1D-CNN)	5.82	5.87	5.86
LSTM+2D-CNN	0.42	0.74	0.74
DL Late Fusion (TLF)	6.24	6.61	6.60

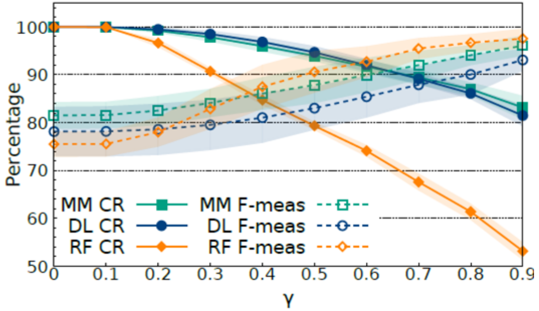
Multimodal-DL shows an **RTPE > 3.5x lower** than its "main competitor" 1D-CNN (L7-784)

MIMETIC: FURTHER GAINS WITH CENSORING



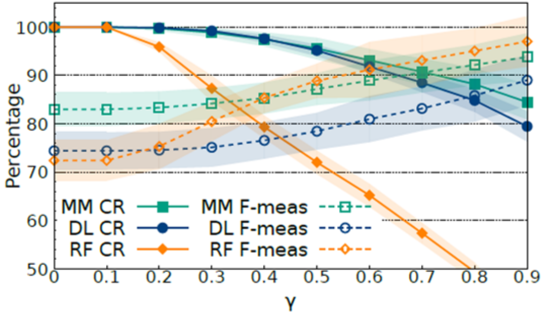
1 FB/FBM



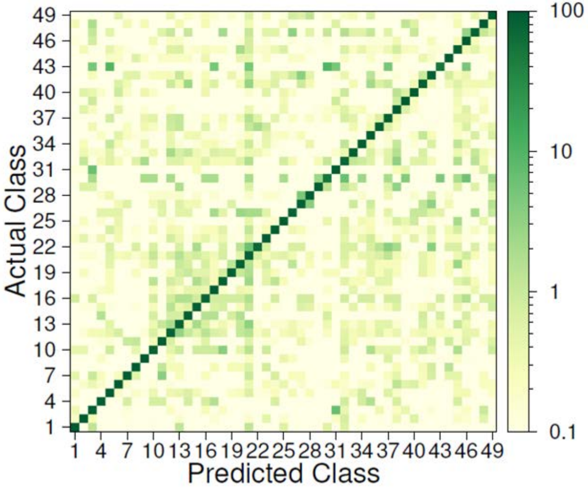


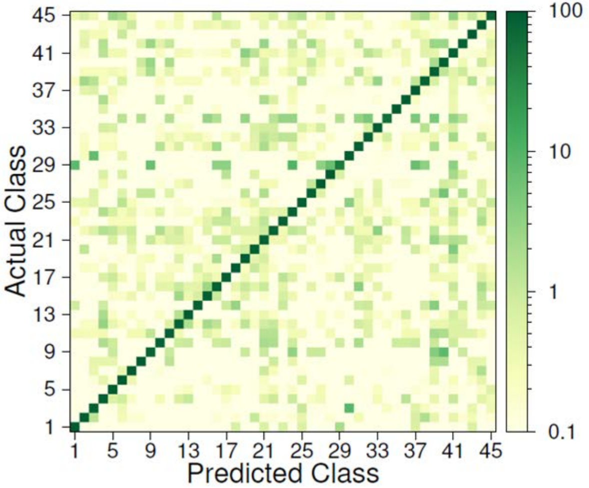
2 Android

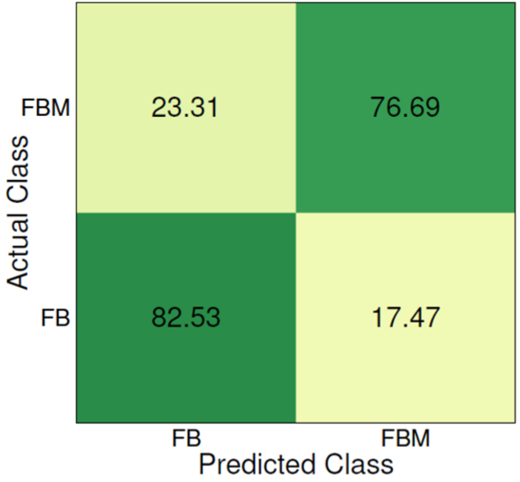
3 iOS



FINE-GRAINED PERFORMANCE IMPROVEMENT

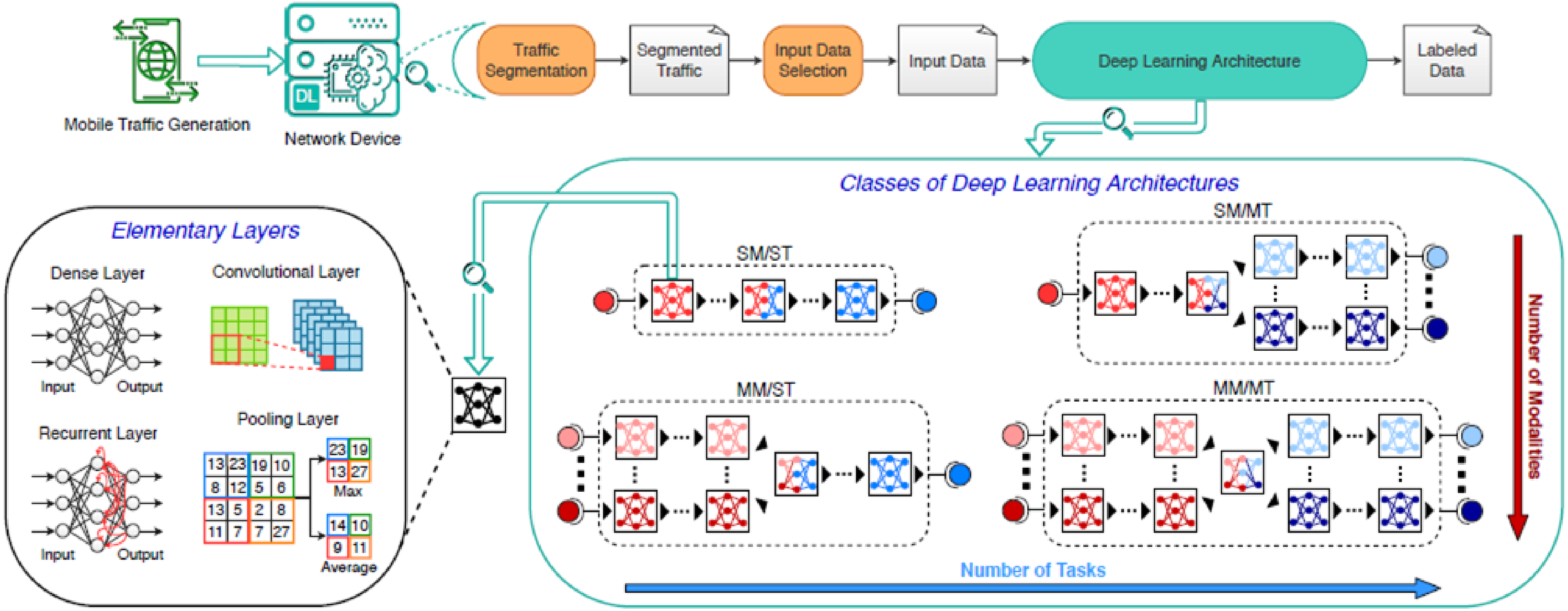






Multimodal-DL achieves **almost-uniform error patterns** in the three cases considered

TOWARD A GENERAL DL-BASED TC FRAMEWORK

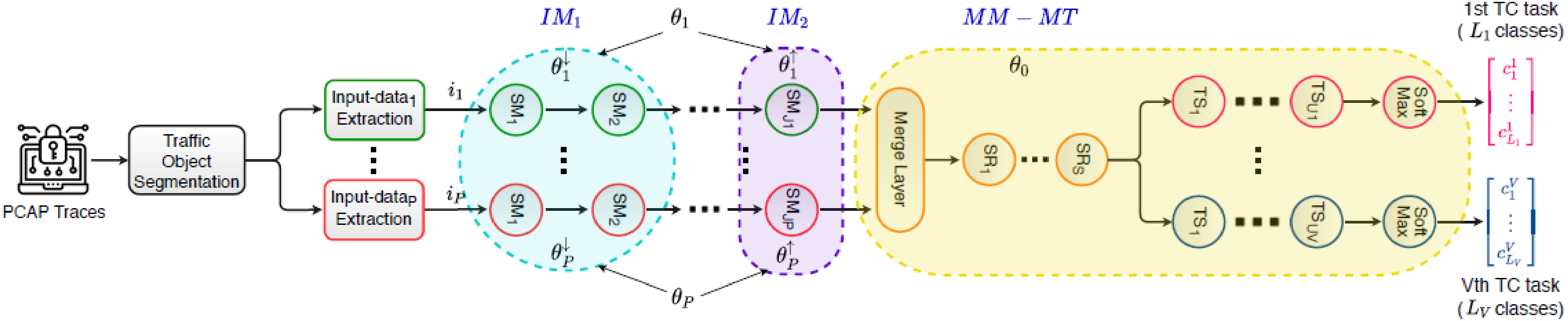


Requirement: Multiple TC desiderata

PUSHING FORWARD: DISTILLER

Deep LearnIng-baSed MulTImodaL MuLtitask EncRypted Traffic Classification

- Capturing both **intra-** and **inter-modalities (multimodal)**
- Able to classify according to **different views (multitask)**

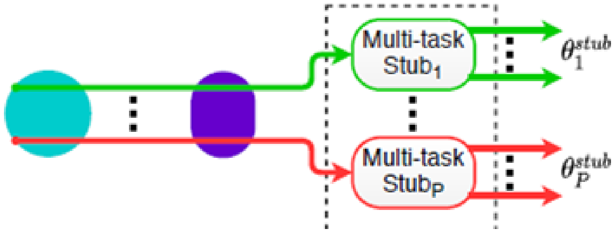


(Architectural Overview)

DISTILLER: FOCUS ON TRAINING

- M : no. of training samples
- V : no. of tasks
- P : no. of modalities

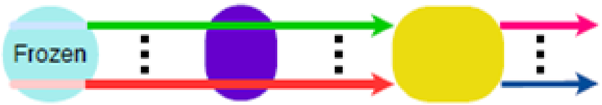
(I) Pre-training



p-th modality loss function

$$\mathcal{L}_p\left(\boldsymbol{\theta}_p, \boldsymbol{\theta}_p^{\text{stub}}\right) \triangleq \sum_{v=1}^{V} \lambda_v \left\{ \sum_{m=1}^{M} \text{CE}(\boldsymbol{t}^v(m),\, \boldsymbol{c}^v(m)\,[\boldsymbol{\theta}_p, \boldsymbol{\theta}_p^{\text{stub}}]) \right\}$$

(II) Fine-tuning

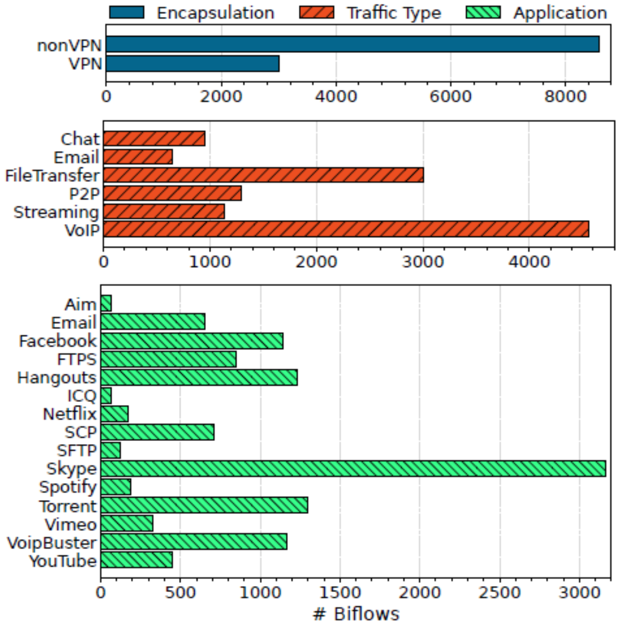


Overall loss function

$$\mathcal{L}\left(\boldsymbol{\theta}_{1:P}^{\uparrow}, \boldsymbol{\theta}_0\right) \triangleq \sum_{v=1}^{V} \lambda_v \sum_{m=1}^{M} \text{CE}(\boldsymbol{t}^v(m),\, \boldsymbol{c}^v(m)[\boldsymbol{\theta}_{1:P}^{\uparrow},\, \boldsymbol{\theta}_0])$$

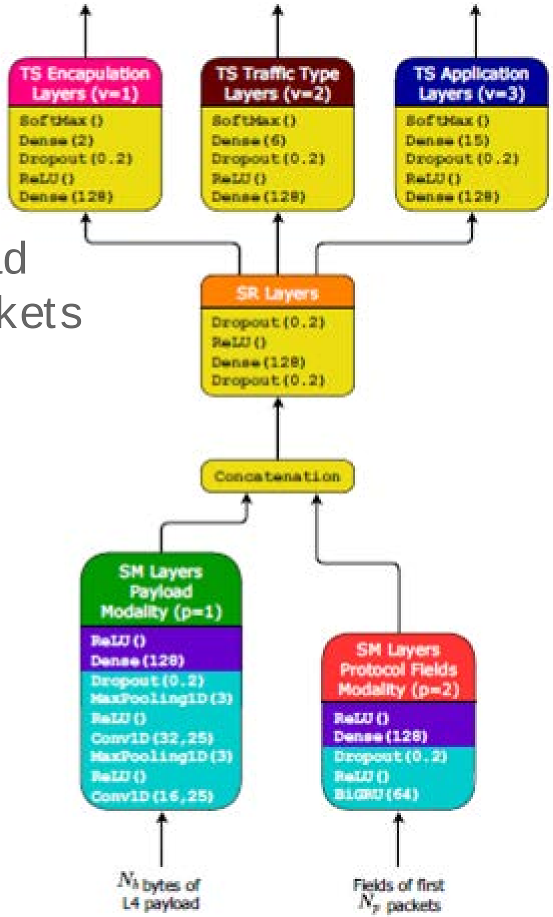
DISTILLER: TAKING ONE INSTANCE

ISCX VPN-nonVPN dataset



- P = 2 Modalities
  - Nb bytes of L4 payload
  - Fields of first Np packets

- V = 3 Tasks:
  - [2] VPN/non-VPN
  - [6] Traffic Type (e.g. P2P, Chat)
  - [15] Applications (e.g. Hangouts)



DISTILLER: PERFORMANCE IN THE MULTI-TASK WILD

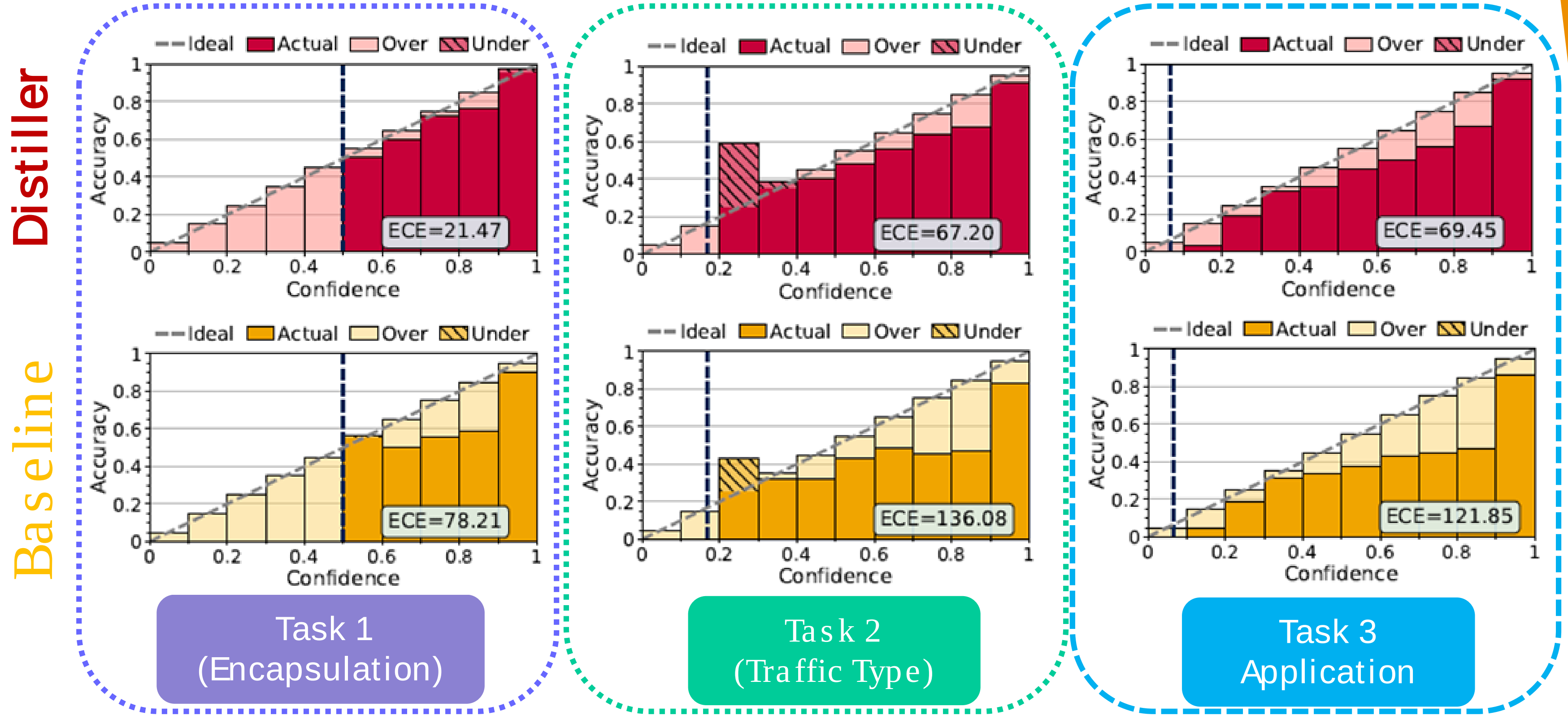
🏆 Overall best classifier

↑ Overall best baseline

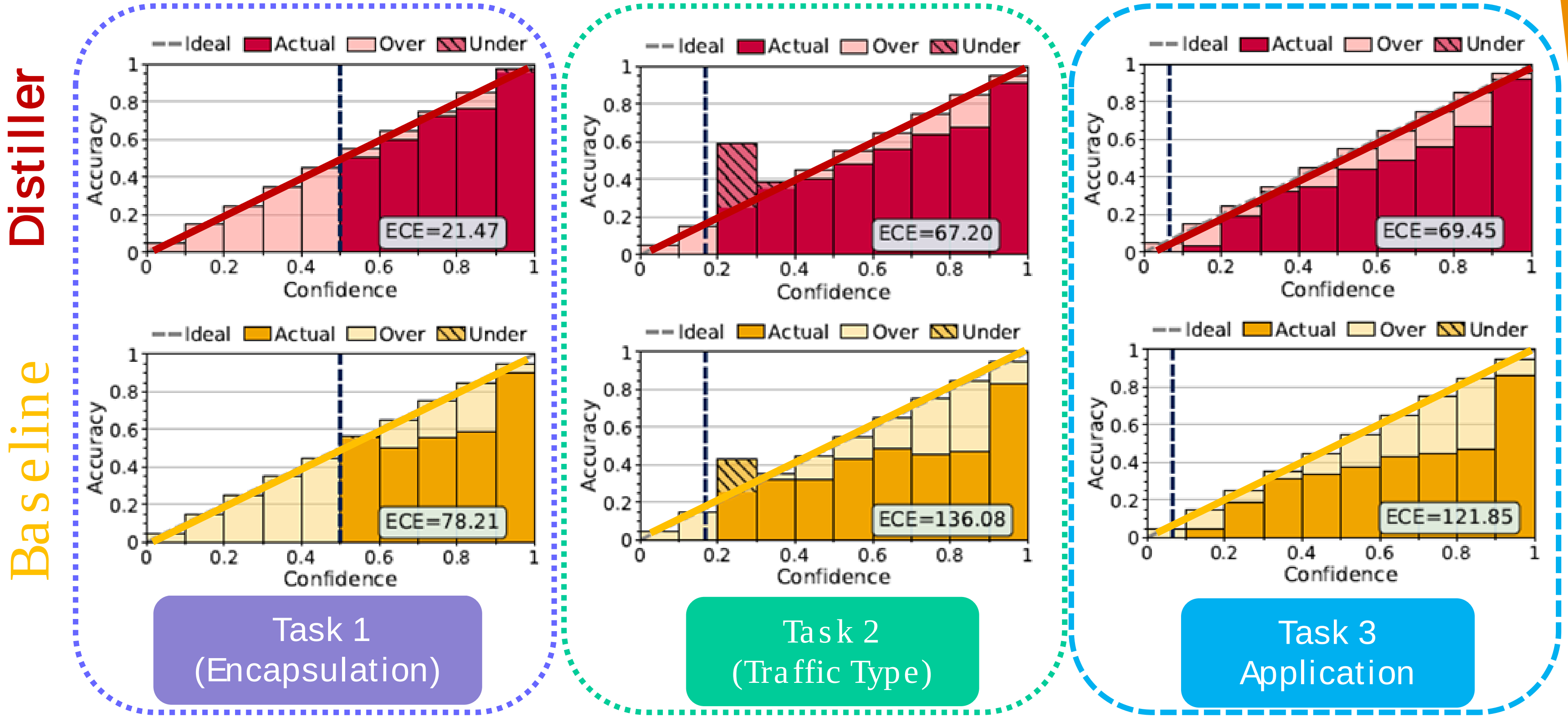
Rank	Multitask Classifier	T1 - Encapsulation Accuracy [%]	T1 - Encapsulation F-measure [%]	T2 - Traffic Type Accuracy [%]	T2 - Traffic Type F-measure [%]	T3 - Application Accuracy [%]	T3 - Application F-measure [%]	RTPE [s]
I	DISTILLER	93.75 (± 0.73) 🏆	91.95 (± 0.67) 🏆	80.78 (± 0.95) 🏆	78.72 (± 1.05) 🏆	77.63 (± 0.66) 🏆	66.44 (± 1.76) 🏆	5.99 (± 0.13)
II	1D-CNN (PAY) [13]	87.47 (± 0.29)	83.50 (± 0.75)	73.14 (± 0.79) ↑	71.14 (± 0.87) ↑	72.73 (± 0.77) ↑	61.35 (± 1.60) ↑	13.83 (± 1.67)
III	2D-CNN (PAY) [20]	87.43 (± 0.66)	83.51 (± 0.46)	71.86 (± 0.95)	69.77 (± 0.96)	71.45 (± 1.13)	59.29 (± 2.06)	40.94 (± 3.57)
IV	MLP (PAY) [26]	86.95 (± 0.65)	82.38 (± 1.12)	70.67 (± 0.64)	68.14 (± 0.72)	69.50 (± 0.97)	56.44 (± 2.45)	2.58 (± 0.36)
V	MLP (HDR) [26]	88.71 (± 0.37) ↑	84.94 (± 0.48) ↑	68.57 (± 0.51)	65.87 (± 0.55)	63.97 (± 1.02)	51.14 (± 1.28)	2.24 (± 0.17)
VI	MLP (PAY) [22]	85.28 (± 0.66)	81.16 (± 0.55)	67.60 (± 1.10)	64.68 (± 1.36)	65.39 (± 1.06)	51.78 (± 1.31)	0.75 (± 0.10) ↑🏆
VII	HYBRID (HDR) [15]	87.11 (± 1.88)	82.82 (± 1.28)	66.00 (± 2.61)	62.40 (± 4.34)	60.17 (± 3.70)	50.49 (± 2.40)	3.34 (± 0.38)
VIII	MLP (HDR) [22]	86.53 (± 0.65)	81.55 (± 1.03)	62.86 (± 0.92)	59.43 (± 1.40)	59.34 (± 0.88)	44.20 (± 1.22)	0.79 (± 0.02)
IX	1D-CNN (HDR) [25]	82.95 (± 1.33)	76.24 (± 2.55)	59.09 (± 3.34)	54.75 (± 2.24)	56.54 (± 2.65)	40.87 (± 2.13)	1.70 (± 0.02)
	DISTILLER Gain	+ 6.28 (± 0.80)	+ 8.45 (± 1.13)	+ 7.65 (± 0.20)	+ 7.58 (± 0.95)	+ 4.90 (± 0.60)	+ 5.09 (± 1.17)	- 7.84 (± 1.67)

best baseline identified

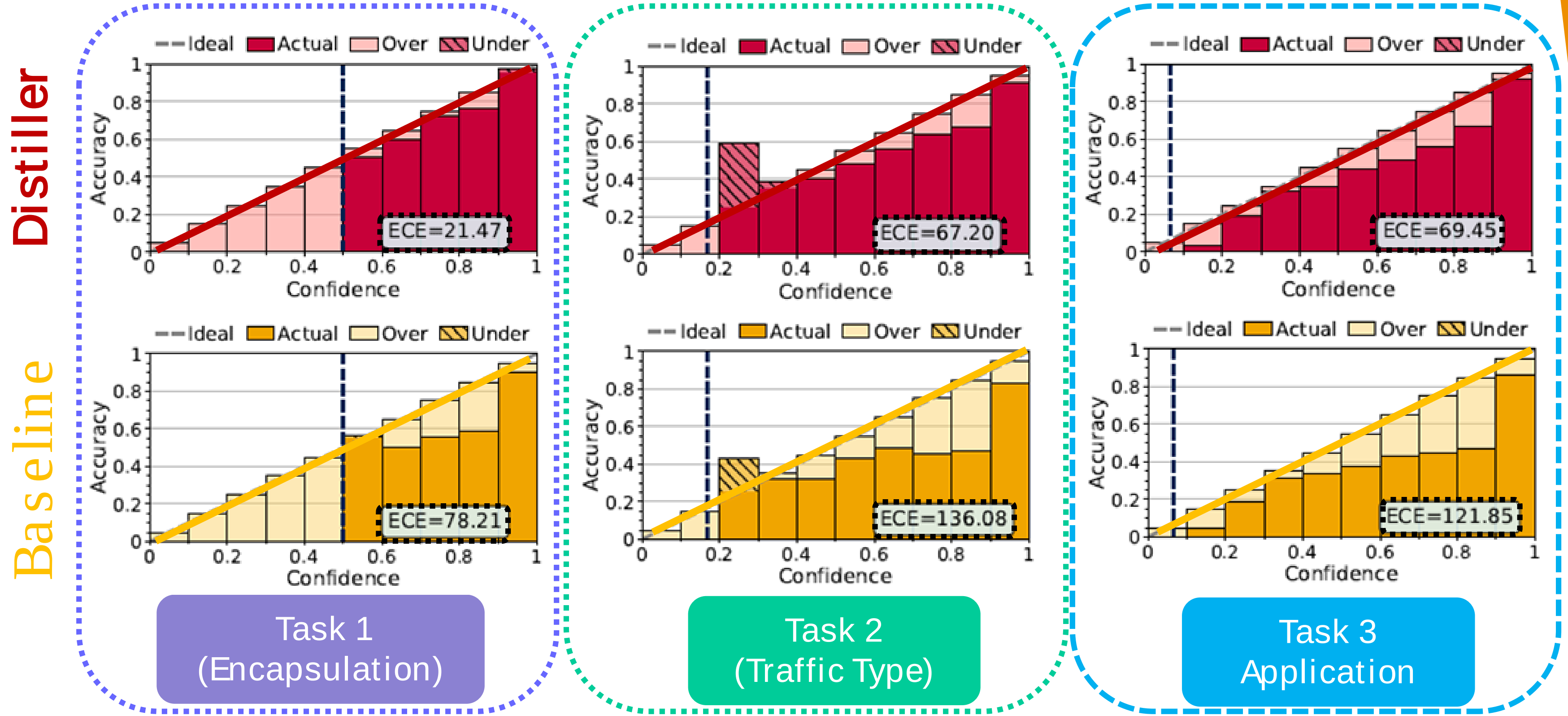
DISTILLER: ACHIEVING BETTER CALIBRATION



DISTILLER: ACHIEVING BETTER CALIBRATION

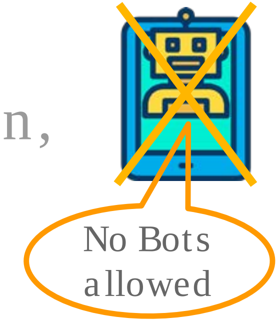


DISTILLER: ACHIEVING BETTER CALIBRATION



BENCHMARKING TC: NEED FOR QUALIFIED DATASETS

Data-driven TC methodologies require **reliably labeled datasets** to ensure proper design, realization, and validation



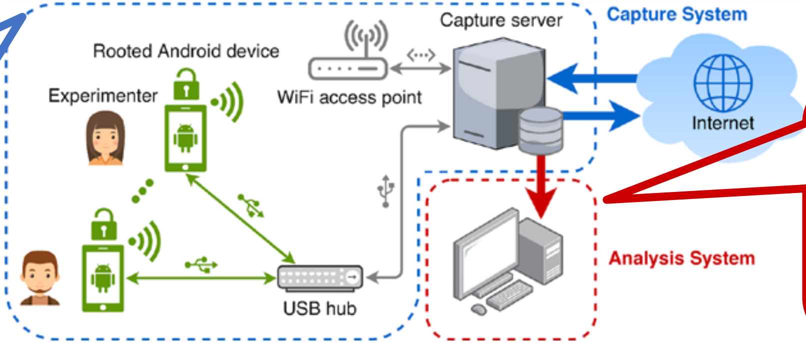
MIRAGE

Reproducible architecture for generating **mobile-app traffic** and automatically creating the related high accurate **ground-truth**

MIRAGE: OVERVIEW

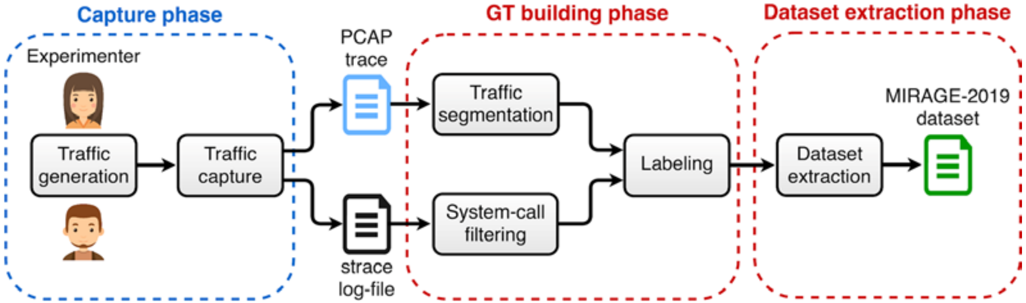
Architecture

- Provides connectivity to mobile devices
- Collects network traffic and system-call log-files
- Can handle multiple devices at the same time



- Performs the Ground-Truth building
- Constructs the final mobile-app traffic dataset
- Extracts the MIRAGE-2019 public version

Functional overview



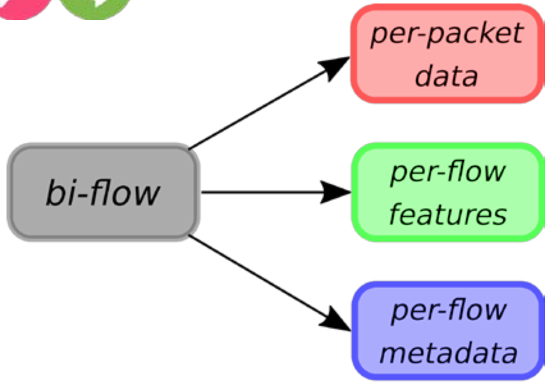
MIRAGE-2019 dataset is available on: http://traffic.comics.unina.it/mirage

MIRAGE IN A NUTSHELL

A **public human-generated** dataset for mobile traffic analysis

- **40** Android apps (no video apps)
- **16** different categories
- **No less than 2500** bi-flows for each app
- Each bi-flow is labeled with the **Android package-name** of generating app

MIRAGE-2019 is released in **JSON** format with information at **different granularities**

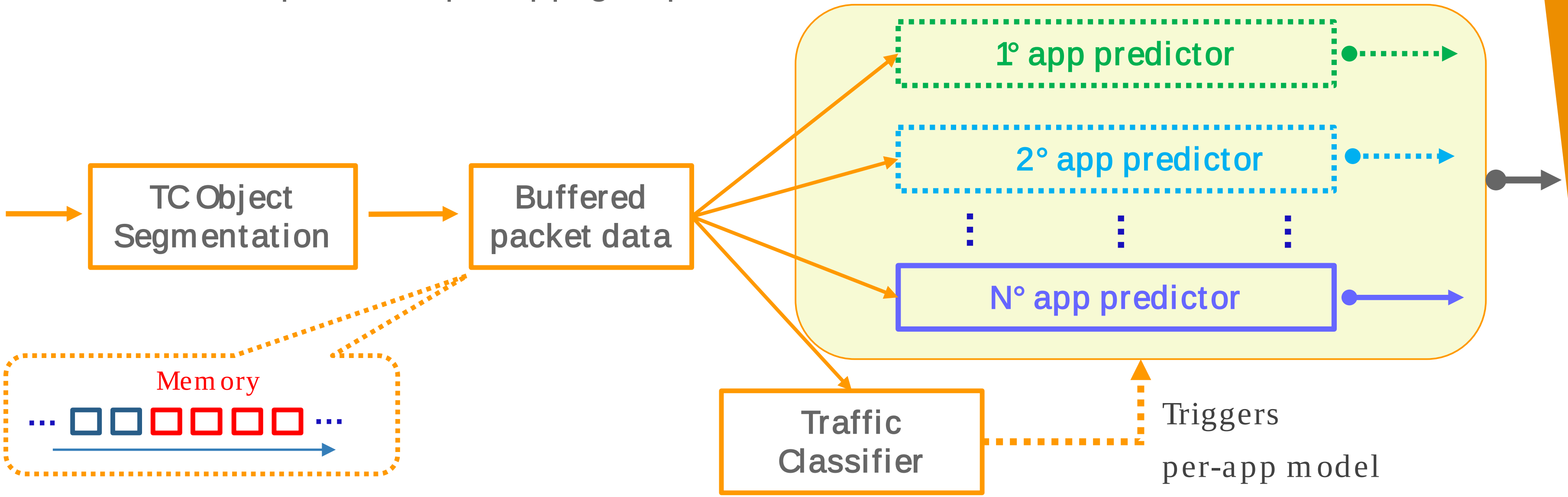


MIRAGE-2019 dataset is available on: http://traffic.comics.unina.it/mirage

MOBILE APPS TRAFFIC PREDICTION

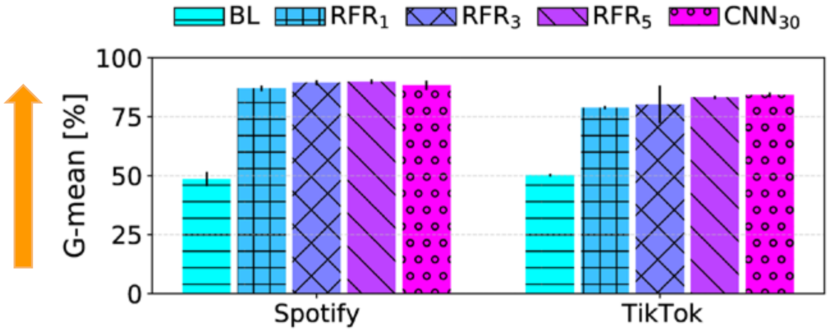
Need for fine-grained network management:

- Traffic is dynamic and of heterogeneous composition
- One predictor for all traffic is not enough
- Idea: One-predictor per app/group

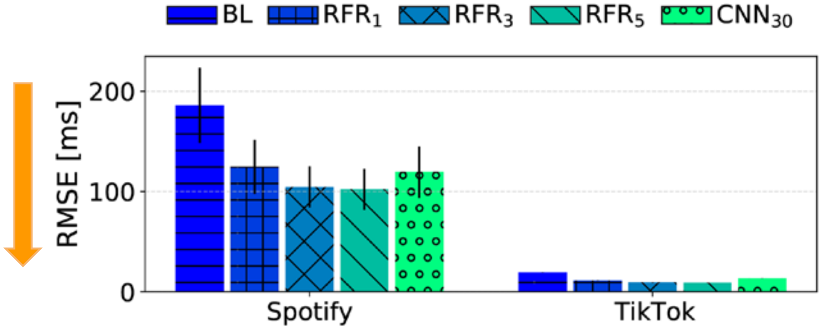


MOBILE TP: INITIAL RESULTS

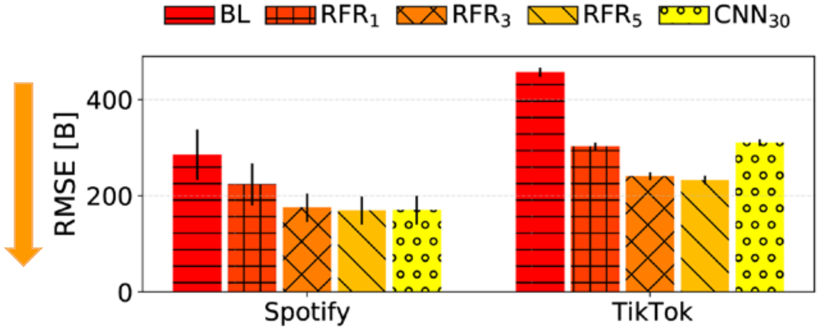
- Direction



- Inter-Arrival Time



- Payload Length



Results come from a 10-fold cross-validation process - Values are shown as μ and σ

LESSON LEARNED AND MILESTONES

DL classifiers fed with raw network traffic data likely lead to misleading performance results

- ✓ Skim informative and unbiased information from input traffic data to DL classifiers

LESSON LEARNED AND MILESTONES

DL classifiers fed with raw network traffic data likely lead to misleading performance results

No “killer” DL architecture for mobile TC

- ✓ Skim informative and unbiased information from input traffic data to DL classifiers
- ✓ Need for advanced hybrid DL architectures with automatically tunable hyper-parameters

LESSON LEARNED AND MILESTONES

DL classifiers fed with raw network traffic data likely lead to misleading performance results

No “killer” DL architecture for mobile TC

Lack of a comprehensive and principled approach to DL-based classifiers applied to mobile TC

- ✓ Skim informative and unbiased information from input traffic data to DL classifiers
- ✓ Need for advanced hybrid DL architectures with automatically tunable hyper-parameters
- ✓ First attempt to the formalization of a comprehensive performance evaluation framework

LESSON LEARNED AND MILESTONES

DL classifiers fed with raw network traffic data likely lead to misleading performance results

No “killer” DL architecture for mobile TC

Lack of a comprehensive and principled approach to DL-based classifiers applied to mobile TC

Lack of general architecture for solving multi-purpose TC tasks with high performance

- ✓ Skim informative and unbiased information from input traffic data to DL classifiers
- ✓ Need for advanced hybrid DL architectures with automatically tunable hyper-parameters
- ✓ First attempt to the formalization of a comprehensive performance evaluation framework
- ✓ A framework for the design of Multimodal Multitask DL Traffic Classifiers

LESSON LEARNED AND MILESTONES

DL classifiers fed with raw network traffic data likely lead to misleading performance results

No “killer” DL architecture for mobile TC

Lack of a comprehensive and principled approach to DL-based classifiers applied to mobile TC

Lack of general architecture for solving multi-purpose TC tasks with high performance

Lack of available datasets for experimentation

- ✓ Skim informative and unbiased information from input traffic data to DL classifiers
- ✓ Need for advanced hybrid DL architectures with automatically tunable hyper-parameters
- ✓ First attempt to the formalization of a comprehensive performance evaluation framework
- ✓ A framework for the design of Multimodal Multitask DL Traffic Classifiers
- ✓ Proposing the MIRAGE project

LESSON LEARNED AND MILESTONES

DL classifiers fed with raw network traffic data likely lead to misleading performance results

No “killer” DL architecture for mobile TC

Lack of a comprehensive and principled approach to DL-based classifiers applied to mobile TC

Lack of general architecture for solving multi-purpose TC tasks with high performance

Lack of available datasets for experimentation

Need for fine-grained prediction of mobile traffic

- ✓ Skim informative and unbiased information from input traffic data to DL classifiers
- ✓ Need for advanced hybrid DL architectures with automatically tunable hyper-parameters
- ✓ First attempt to the formalization of a comprehensive performance evaluation framework
- ✓ A framework for the design of Multimodal Multitask DL Traffic Classifiers
- ✓ Proposing the MIRAGE project
- ✓ Investigating DL-based biflow-level app-tailored predictors

ESSENTIAL REFERENCES

1. **“The applications of deep learning on traffic identification”,** BlackHat USA, 2015
2. **“Appscanner: Automatic fingerprinting of smartphone apps from encrypted network traffic”,** IEEE EuroS& P 2016
3. **“End-to-end encrypted traffic classification with one-dimensional convolution neural networks”,** IEEE ISI 2017
4. **“Malware traffic classification using convolutional neural network for representation learning”,** IEEE ICIN, 2017
5. **“Network traffic classifier with convolutional and recurrent neural networks for Internet of Things”,** IEEE Access, 2017
6. **“Deep packet: A novel approach for encrypted traffic classification using deep learning”,** Soft Computing, 2020
7. **“Automatic multi-task learning system for abnormal network traffic detection”,** International Journal of Emerging Technologies in Learning, 2018
8. **“Common knowledge based and one-shot learning enabled multi-task traffic classification”,** IEEE Access 2019
9. **“FS-Net: A flow sequence network for encrypted traffic classification”,** IEEE INFOCOM, 2019
10. **“Multi-task network anomaly detection using federated learning”,** ACM SoICT, 2019

...AND OUR REFERENCES

1. **"PortLoad: taking the best of two worlds in traffic classification"**, IEEE INFOCOM Workshops, 2010
2. **"Multi-Classification Approaches for Classifying Mobile App Traffic"**, Elsevier Journal of Network and Computer Applications, 2018
3. **"Mobile Encrypted Traffic Classification Using Deep Learning: Experimental Evaluation, Lessons Learned, and Challenges"**, IEEE Transactions on Network and Service Management, 2019
4. **"MIMETIC: Mobile Encrypted Traffic Classification using Multimodal Deep Learning"**, Elsevier Computer Networks, 2019 (BEST PAPER AWARD)
5. **"Toward Effective Mobile Encrypted Traffic Classification through Deep Learning"**, Elsevier Neurocomputing, 2020
6. **"DISTILLER: Encrypted Traffic Classification via Multimodal Multitask Deep Learning"**, Elsevier Journal of Network and Computer Applications, 2020 (submitted)
7. **"Characterization and Prediction of Mobile-App Traffic using Markov Modeling"**, IEEE Transactions on Network and Service Management, 2020 (submitted)
8. **"Know your Big Data Trade-offs when Classifying Encrypted Mobile Traffic with Deep Learning"**, IEEE/ACM TMA 2019
9. **"MIRAGE: Mobile-app Traffic Capture and Ground-truth Creation"**, 4th IEEE ICCCS 2019 (BEST PAPER AWARD)

Thank you!

Questions?

domenico.ciuonzo@unina.it

domenicociuonzo.wordpress.com

traffic.comics.unina.it

DIETI. UNIVERSITA' DEGLI STUDI DI NAPOLI FEDERICO II