Few Shot Learning Approaches for Classifying Rare Mobile-App Encrypted Traffic Samples

Giampaolo Bovenzi, Davide Di Monda, Antonio Montieri,
Valerio Persico, and Antonio Pescapè

University of Napoli Federico II, Napoli (Italy)

AI per l'Industria
Ital-IA, Pisa 29-31 May 2023

Mobile-App Traffic Classification

- **Deep Learning (DL)** is **effective** for classifying **encrypted network traffic**
  - But it requires **large amounts of labeled data** to provide satisfactory results
- Collection of **large labeled network-traffic datasets**
- Number of **new apps constantly rising**
  (4.67 million apps during the last quarter of 2021[1])

[1] Number of available apps in the Google Play Store from 2nd quarter 2015 to 2nd quarter 2022 - www.statista.com

Mobile-App Traffic Classification: Issues

- **Deep Learning (DL)** is **effective** for classifying **encrypted network traffic**
  - But it requires **large amounts of labeled data** to provide satisfactory results
- Collection of **large labeled network-traffic datasets**
  - Time-consuming process
  - User-privacy and business-sensitivity concern
- Number of **new apps constantly rising**
  (4.67 million apps during the last quarter of 2021[1])
  - DL models need to be re-trained in order to classify the newly published apps

[1] Number of available apps in the Google Play Store from 2nd quarter 2015 to 2nd quarter 2022 - www.statista.com

Mobile-App Traffic Classification: Issues

- **Deep Learning (DL)** is **effective** for classifying **encrypted network traffic**
  - But it requires **large amounts of labeled data** to provide satisfactory results
- Collection of **large labeled network traffic datasets**
  - Time-consuming process
  - User-privacy and business-sensitivity concern
- Number of **new apps constantly rising**
(4.67 million apps during the last quarter of 2021[1])
  - DL models need to be re-trained in order to classify the newly published apps

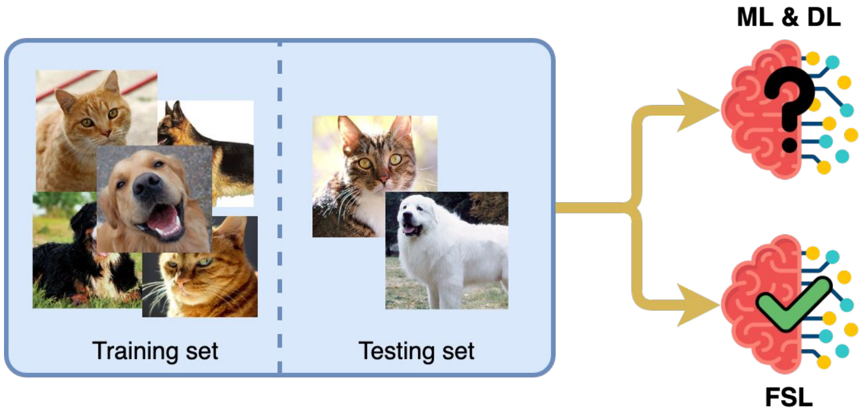
Often only a
few labeled traffic samples
are available

Adoption of
DL falls short

[1] Number of available apps in the Google Play Store from 2nd quarter 2015 to 2nd quarter 2022 - www.statista.com

Few-Shot Learning

Few-Shot Learning (FSL) aims at tackling this issues, by leveraging non-few knowledge (**prior knowledge**) in order to build a **model capable of generalizing** enough on new tasks **with few samples available**



Research questions

- Can be **FSL approaches** applied to the **mobile-app encrypted traffic classification**?
  - … and how to tailor it to this domain?
- What is the impact of using different **FSL setups** in terms of **number of training classes N** (viz. Apps) and **number of shots K** (viz. biflow for each App)?

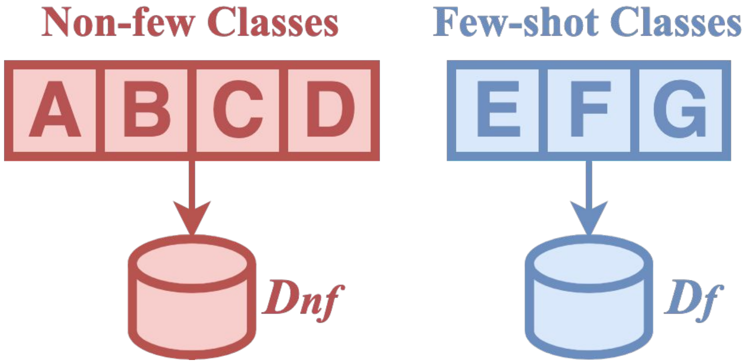
N-way K-shot setup

Few-Shot Learning: Paradigms

- **Transfer Learning**
  - Aims to *transfer knowledge* from a task to a related one with the objective of **fast adaptation**, **reduced complexity**, and **performance improvements**
- **Meta Learning**
  - It is the ability of "**Learning to learn**" or *learning to compare*
  - The ultimate goal is to provide a **model capable of generalizing** enough on tasks with **unseen** classes

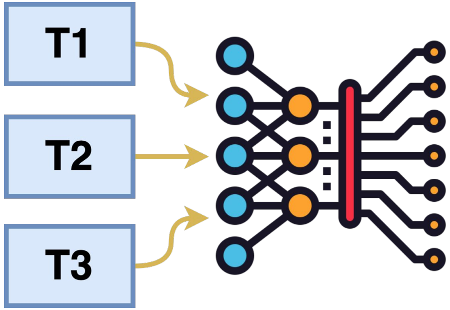
Preliminary: Dataset Partitioning

- The **most populated** classes are *separated from* the **less populated** ones
  - Most populated classes are included in the training set D_{nf} → used for **training**
  - Less populated classes are included in the testing set D_f → used for **testing**



Transfer-Learning Approaches

- The Transfer-learning approaches use prior knowledge (D_{nf}) to learn a **good initialization point** for the model weights, i.e. *base model*
- The base model is **adapted** to *classify few-shot classes* (D_f)
  - Done via **fine-tuning** to different extents

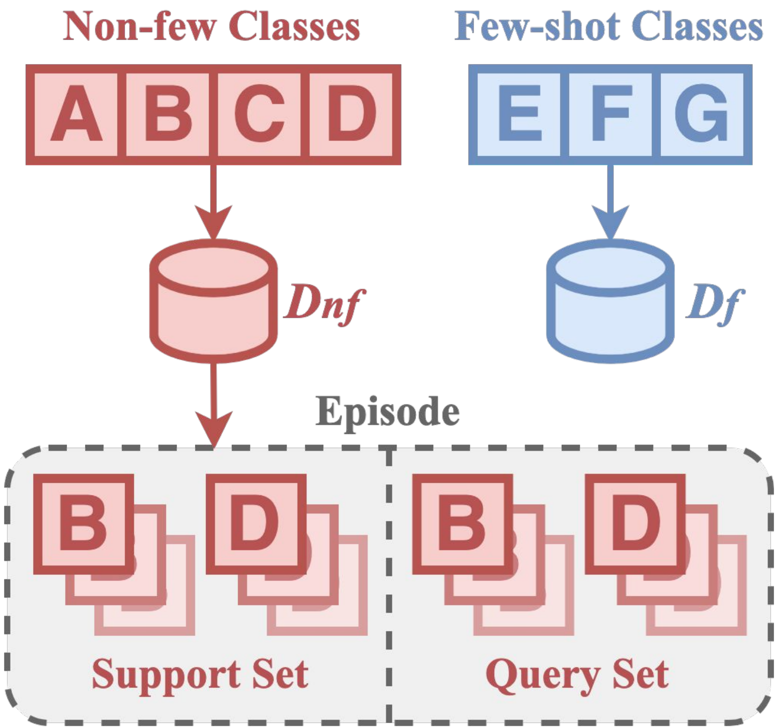


Meta Learning: Episodic Learning

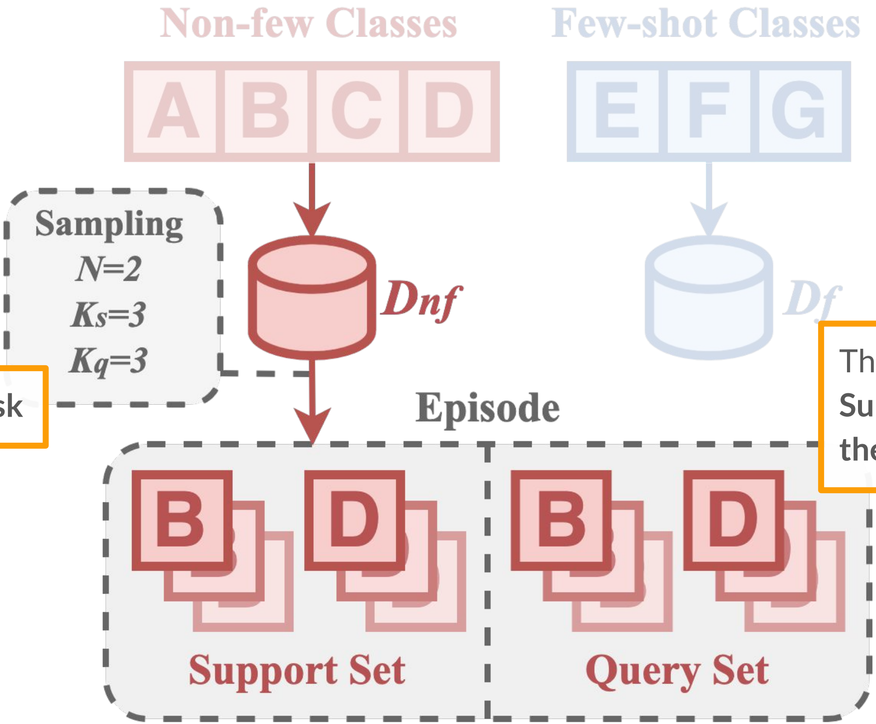
The **meta learning** is **used jointly** with **episodic learning**

Episodic Learning

- Training is organized as series of learning problems (**episodes**)
- Episodes mimic the **inference** scenario



Meta Learning: Task Configuration

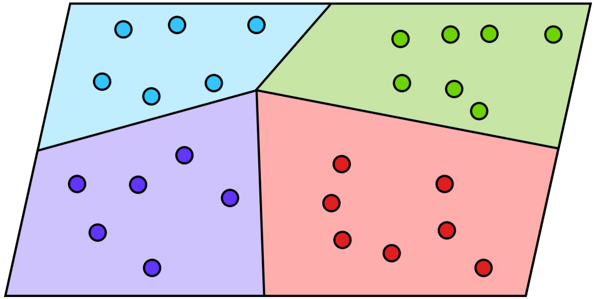


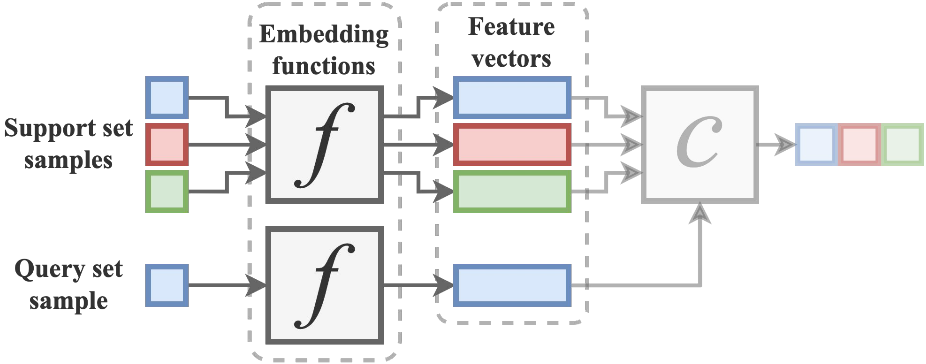
N-Way K-Shot Task

The model **learns** from the **Support Set**, and **measures the error** on the **Query Set**

Meta-Learning Approaches

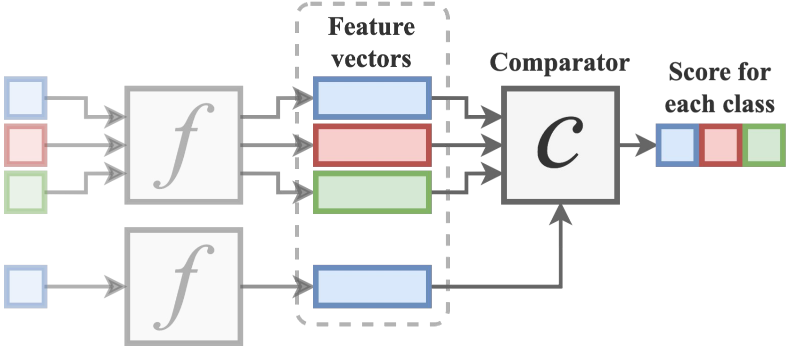
- Prior knowledge (**D**$_{nf}$) used to train an **embedding function (*f*)**
  - *similar* samples are *closer* to each other
  - *dissimilar* samples are more *easily separable*
- Doing so they manage **to reduce the hypothesis space complexity**





Meta-Learning Approaches

- Classification is performed by measuring the similarity of support and query feature vectors through a **comparator (c)**, e.g., k-NN, SVM, NN.
- The output of the comparator is a **similarity score**
- Model-based methods **differ** according to the comparator



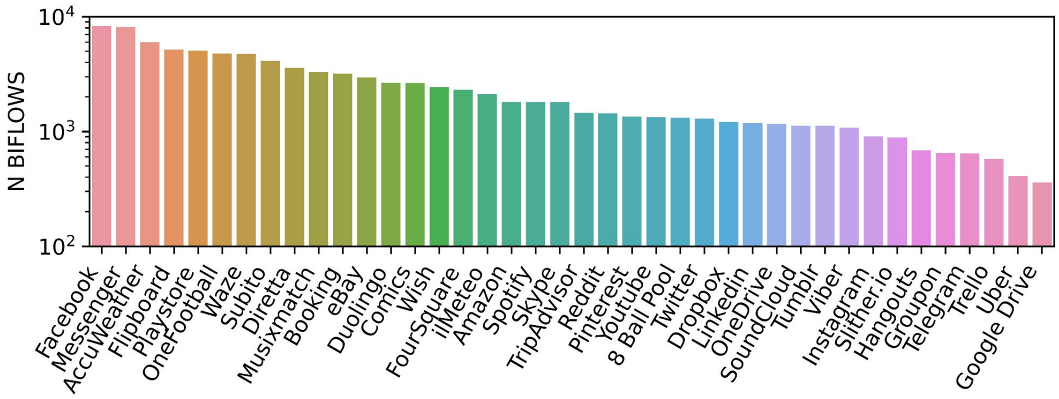
Experimenting with FSL: Dataset

- MIRAGE-2019
  - Collected at ARCLAB University of Napoli Federico II from May ‘17 to May ‘19
  - Publicly available (scan the code!)
  - **Human-generated** dataset (~300 users)
  - **40** popular Android / **16** different app categories
  - **Biflows** as traffic object

MIRAGE-2019 dataset is available at

https://traffic.comics.unina.it/mirage

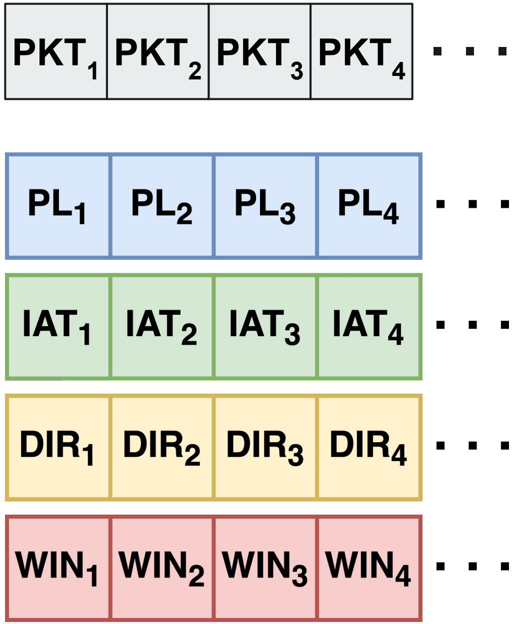
Experimenting with FSL: Dataset Partitioning



D_{nf1}	D_{nf2}	D_f
training	*validation*	*testing*
24 classes top-24 w/ #BF>1e3	8 classes last-8 w/ #BF>1e3	8 classes #BF<1e3

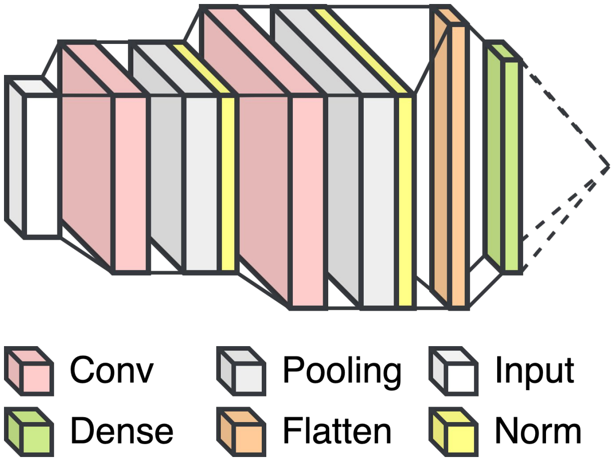
Experimenting with FSL: Input Data

- **PSQ**: informative fields of the first N_P= 10 packets of each biflow
  - (L4) Payload Length (**PL**)
  - Inter-Arrival Time (**IAT**)
  - Direction (**DIR**): *upstream/downstream*
  - TCP Rcv Window (**WIN**): 0 for UDP packets



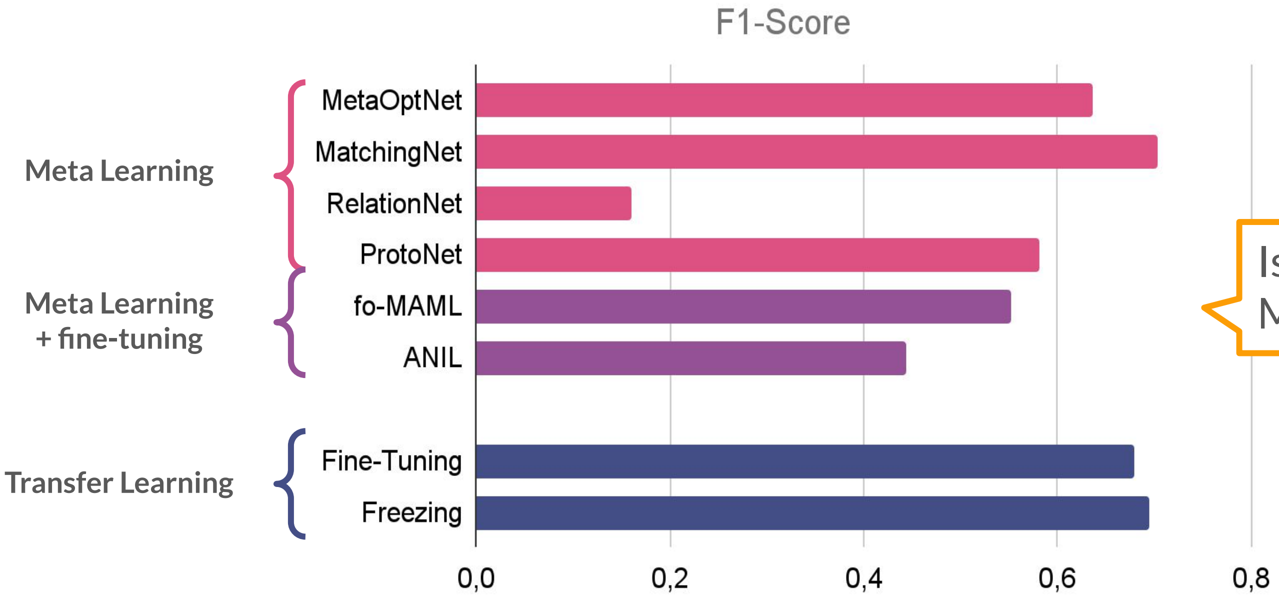
Experimenting with FSL: Embedding Function

2D-CNN
[Lopez et al, IEEE Access, 2017]
Input: ***PSQ***



FSL approaches for Mobile-app Encrypted TC

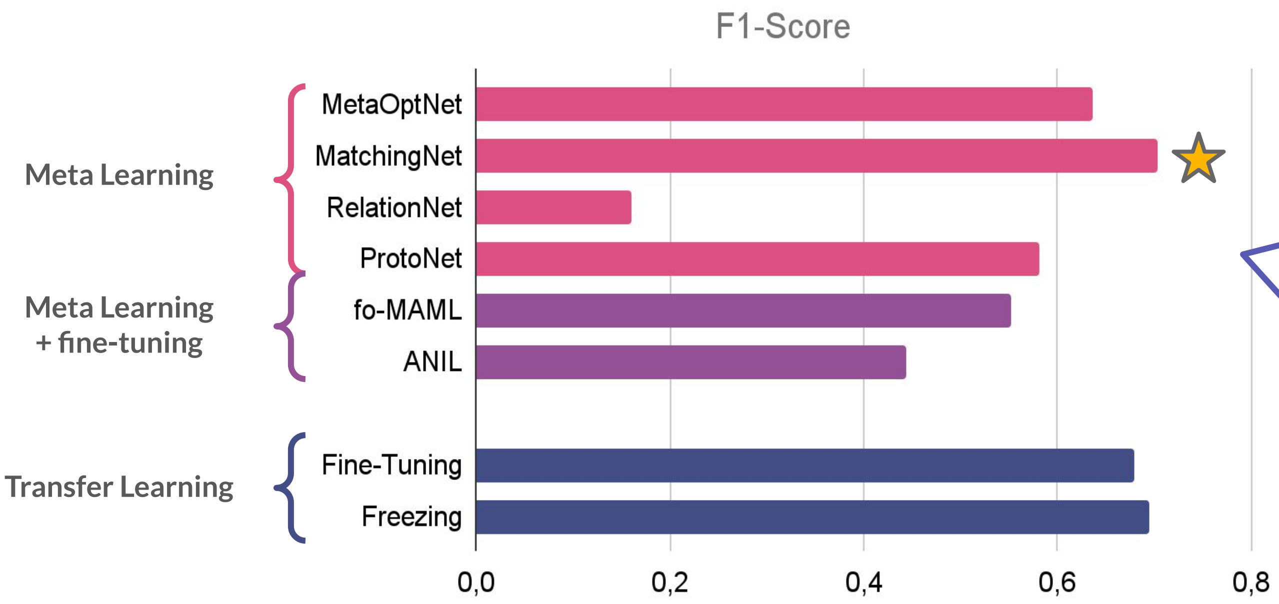
8-way 25-shot



Is **FSL applicable** to Mobile-app Encrypted TC?

FSL approaches for Mobile-app Encrypted TC

8-way 25-shot

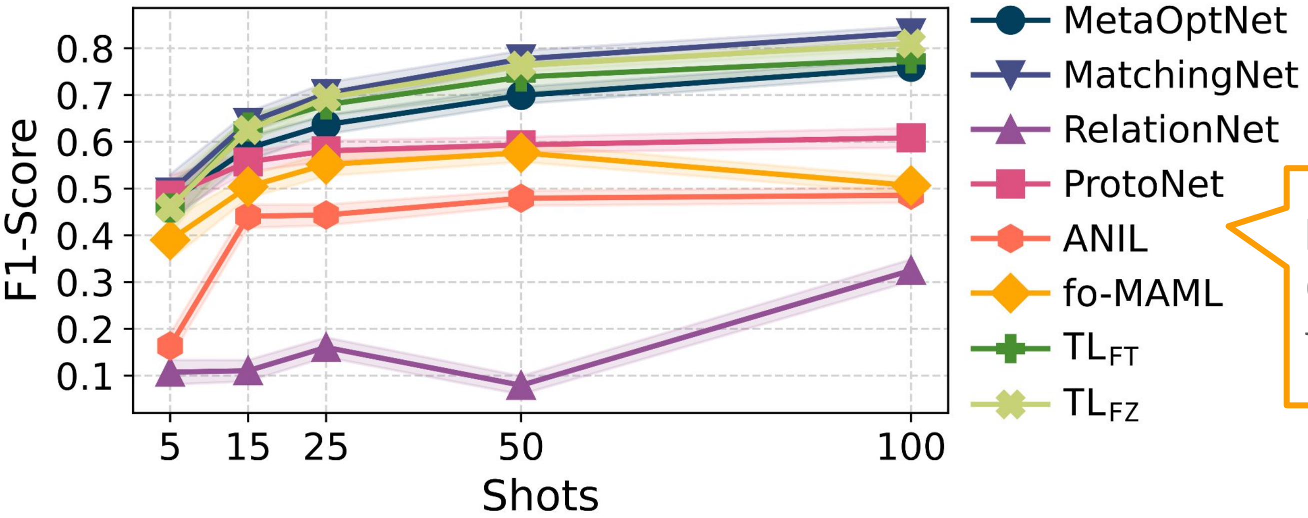


FSL approaches show **variable performances:**

- **Transfer Learning** achieves **good results**
- **RelationNet** and **ANIL** are **under-performing**
- **MatchingNet** is the **best performing**

Impact of the number of Biflows

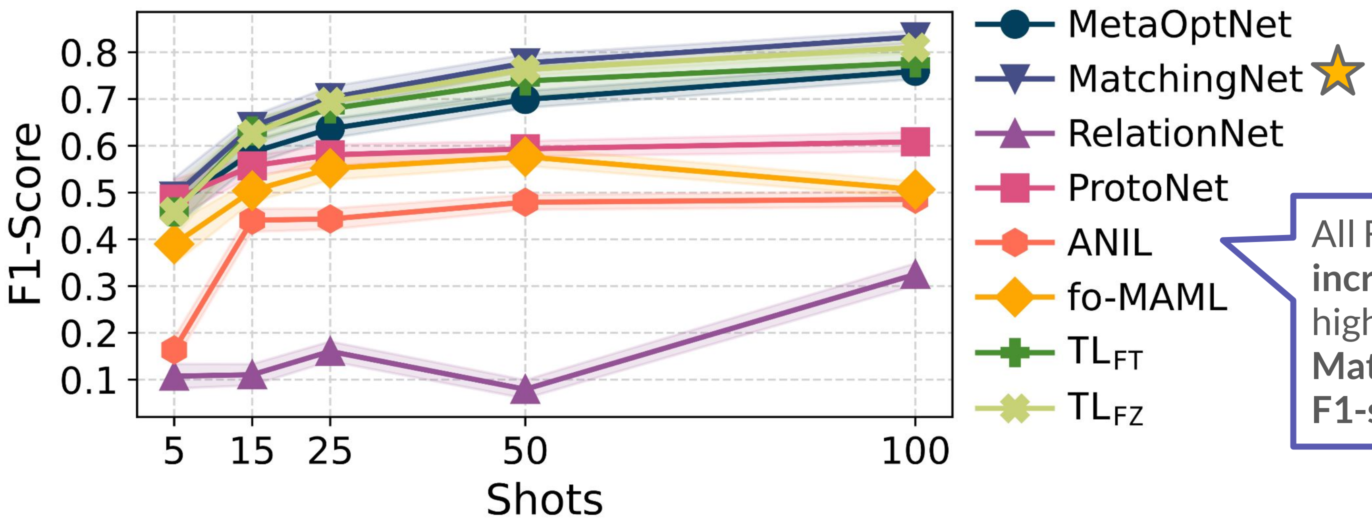
8-way K-shot



How many biflows (shots) per App we need for classify the rare Apps?

Impact of the number of Biflows

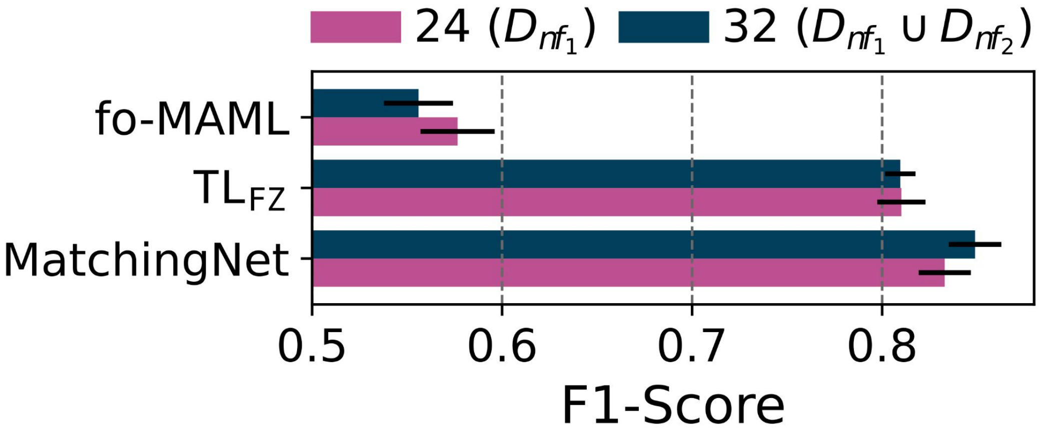
8-way K-shot



All FSL approaches **significantly increase their F1-score** for higher shots.
MatchingNet achieves an **F1-score > 80%** when K = 100

Impact of a wider App pool during training

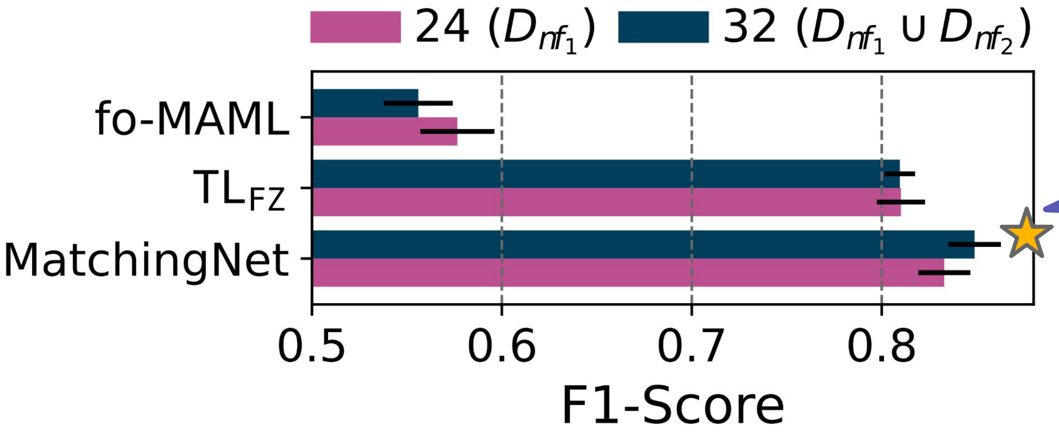
8-way 100-shot



We want to evaluate the ability of the algorithms to extend acquired knowledge on minority classes by using a **wider train class pool**

Impact of a wider App pool during training

8-way 100-shot



MatchingNet performance **improves** when passing from 24 to 32 apps

Ongoing and Future Directions

- **Optimization of the learning objective** by using more complex loss functions to enhance the goodness of embeddings
- The adoption of **different embedding functions** (e.g., **multimodal architectures**) to explore their benefits in this context
- The investigation of **data-based approaches** with the augmentation of samples from few-shot classes

Thanks for your attention

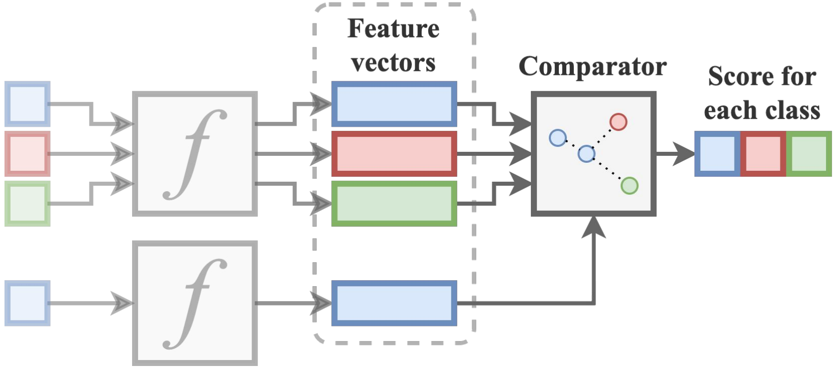
Questions?

Contacts: giampaolo.bovenzi@unina.it - wpage.unina.it/giampaolo.bovenzi

Backup Slides

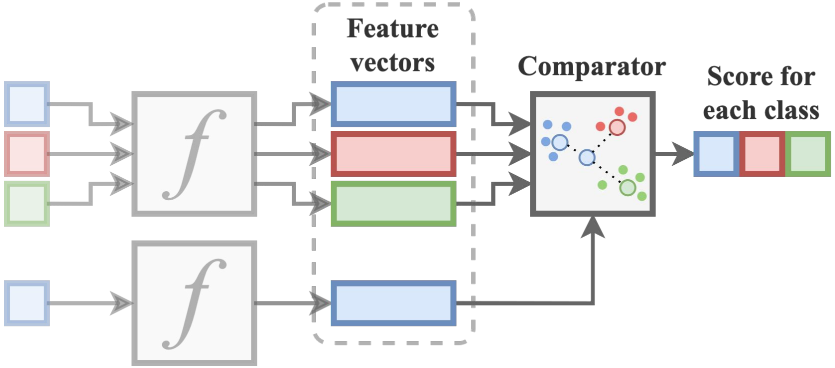
Meta-Learning Approaches

- Matching Networks
    - Distance between the query and support set samples in the embedded space (**nearest-neighbor** based)



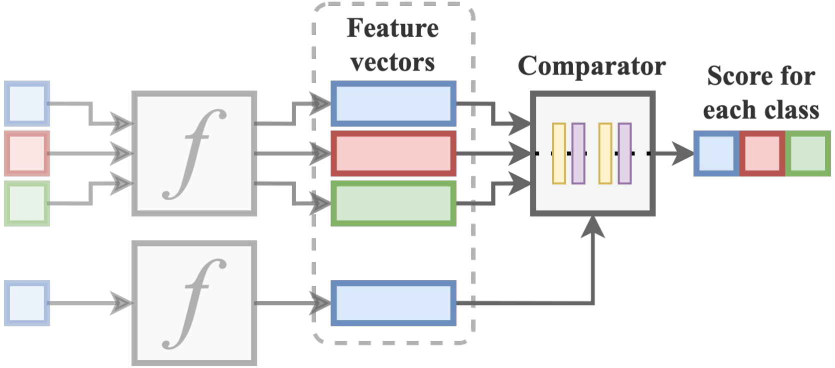
Meta-Learning Approaches

- Matching Networks
- Prototypical Networks
  - Distance between the query sample and *prototypes* of each class in the support set in the embedded space (**nearest-neighbor** based)



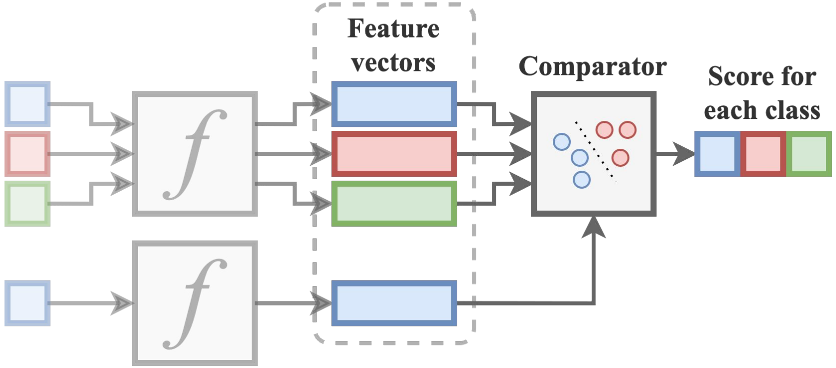
Meta-Learning Approaches

- Matching Networks
- Prototypical Networks
- Relational Networks
    - Measure the similarity in the embedded space between the query sample and prototypes of each class in the support set through a **CNN** with a **Sigmoid Function**



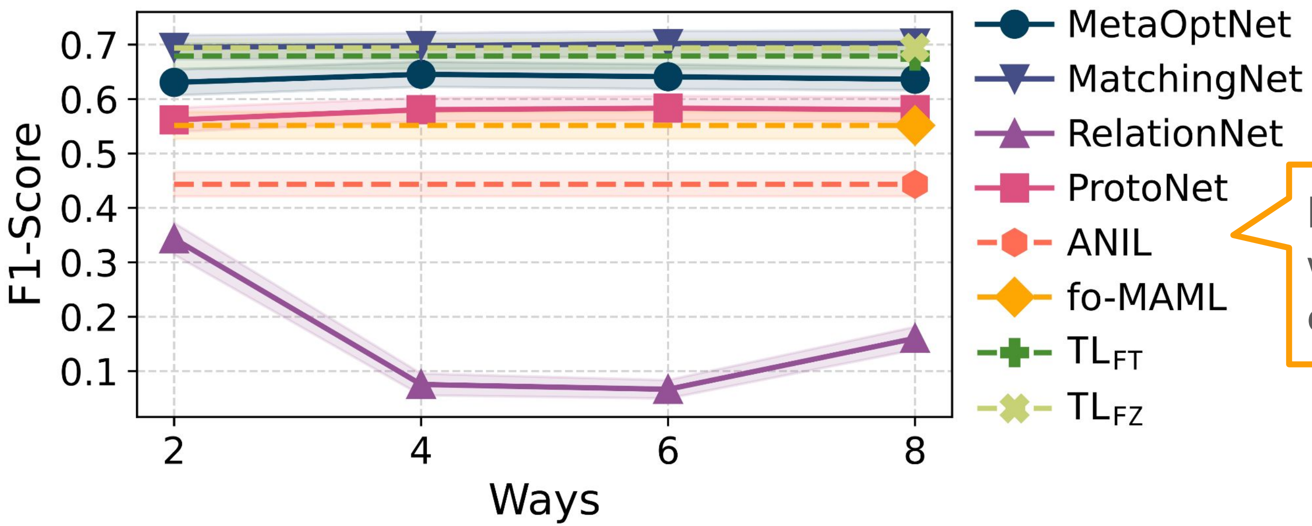
Meta-Learning Approaches

- Matching Networks
- Prototypical Networks
- Relational Networks
- MetaOptNet
    - Employs a linear **Support Vector Machine** (**SVM**) as a base learner



Impact of the number of Apps

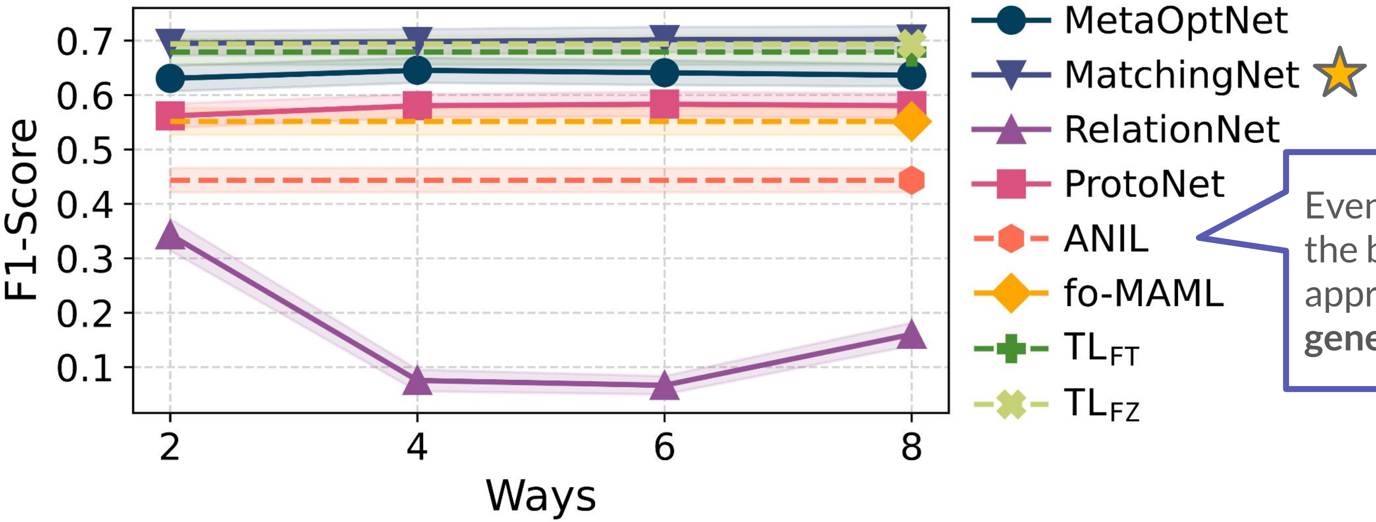
N-way 25-shot



How many Apps (ways) we need to be able to classify rare Apps?

Impact of the number of Apps

N-way 25-shot



Even with just **2 ways tasks** the best performing approaches offer **satisfactory generalization ability**