Graph Neural Networking Challenge 2021

José Suárez-Varela

Barcelona Neural Networking center

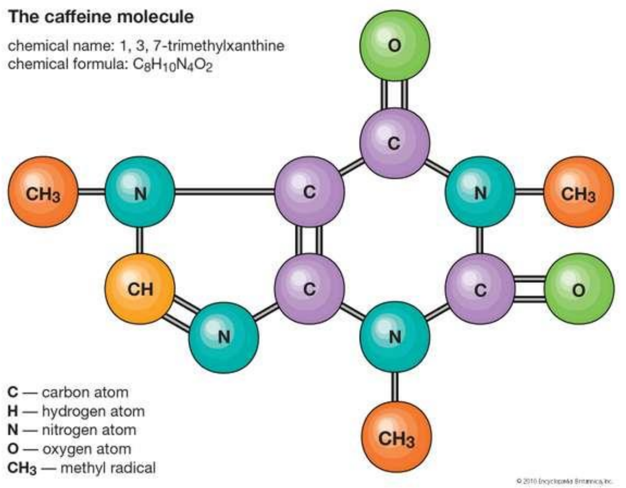
Universitat Politècnica de Catalunya

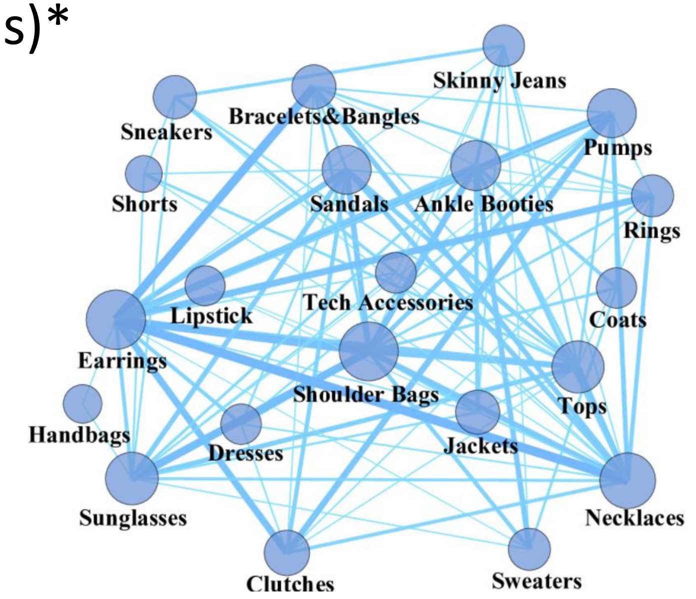
May 21st 2021

What are
Graph Neural Networks?

What are Graph Neural Networks?

- Graph Neural Networks (GNN) is a neural network family tailored to **learn from graph-structured data**
- Extensively used in other fields where data is fundamentally represented as graphs (e.g., chemistry, physics, biology, recommender systems)*





Source: "Dressing as a Whole: Outfit Compatibility Learning Based on Node-wise Graph Neural Networks", WWW conference, 2019

* "Must-read papers on GNN", https://github.com/thunlp/GNNPapers

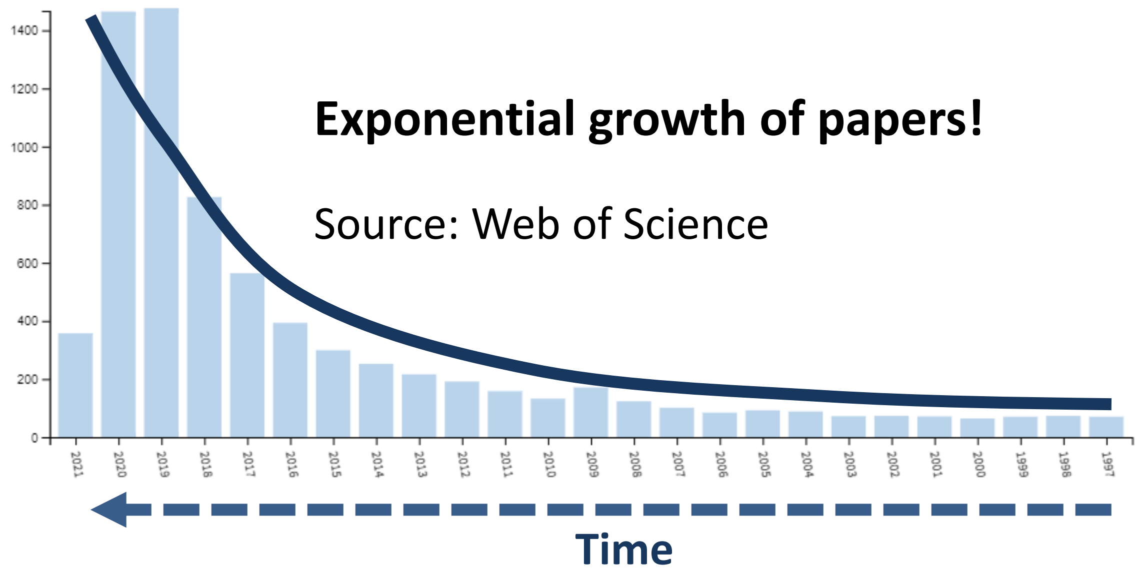
GNNs are the next big thing in AI/ML

- *“Machine Learning on graphs becomes a first-class citizen at AI conferences”* (NeurIPS 2019) [1]

- Top trends in Graph Machine Learning in 2020. “New cool applications of GNN” [2]

- Transformers (the new revolution in the Natural Language Processing field) are Graph Neural Networks [3]

- **Growing number of GNN applications in the networking field [4]**



[1] https://medium.com/mlreview/machine-learning-on-graphs-neurips-2019-875eecd41069
[2] https://towardsdatascience.com/top-trends-of-graph-machine-learning-in-2020-1194175351a3
[3] https://towardsdatascience.com/transformers-are-graph-neural-networks-bca9f75412aa
[4] Must-read papers on GNN for communication networks. https://github.com/BNN-UPC/GNNPapersCommNets

How can Graph Neural Networks be applied to networking?

What is a digital twin?

- A digital twin is a virtual replica of a physical object or process
- It enables to simulate the behavior of a physical system under certain input conditions:
  - What will happen if there is a specific failure? (e.g., in the electrical system)
  - What happen if I make a change in the object? (e.g., new wing design)

Digital twins for networks

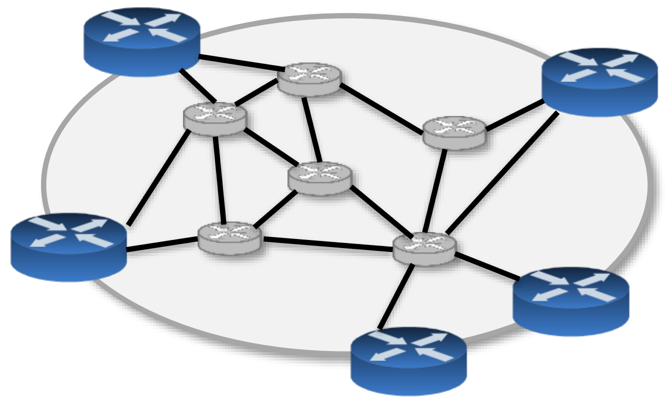
- A Network Digital Twin is fundamentally this box:

Network state (input conditions):
- Topology
- Traffic
- Configuration
- …

Network Digital Twin

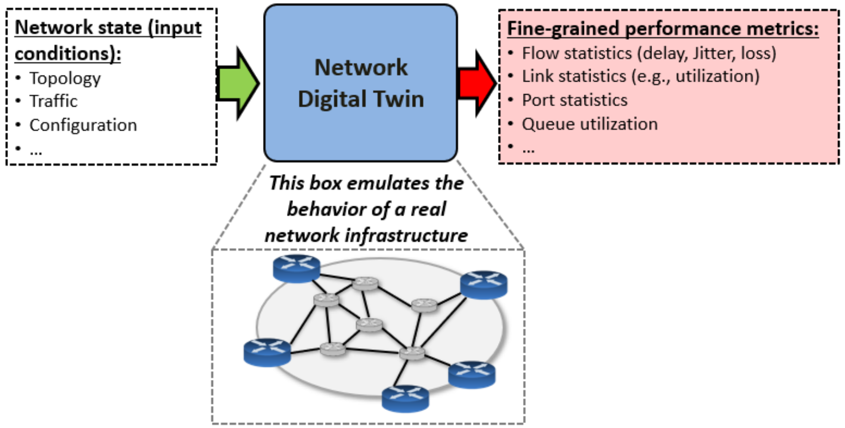
Fine-grained performance metrics:
- Flow statistics (delay, Jitter, loss)
- Link statistics (e.g., utilization)
- Port statistics
- Queue utilization
- …

This box emulates the behavior of a real network infrastructure



Digital Twins can be applied to many fundamental networking use cases

(with especial focus on fine-grained QoS metrics)



Network Planning and Upgrading

- Which is the best link upgrade given a limited budget?
- How much can customer traffic increase until a network upgrade is needed?
- What is the optimal link redundancy to maintain current SLAs considering up to two random link failures?

Troubleshooting and Performance Analysis

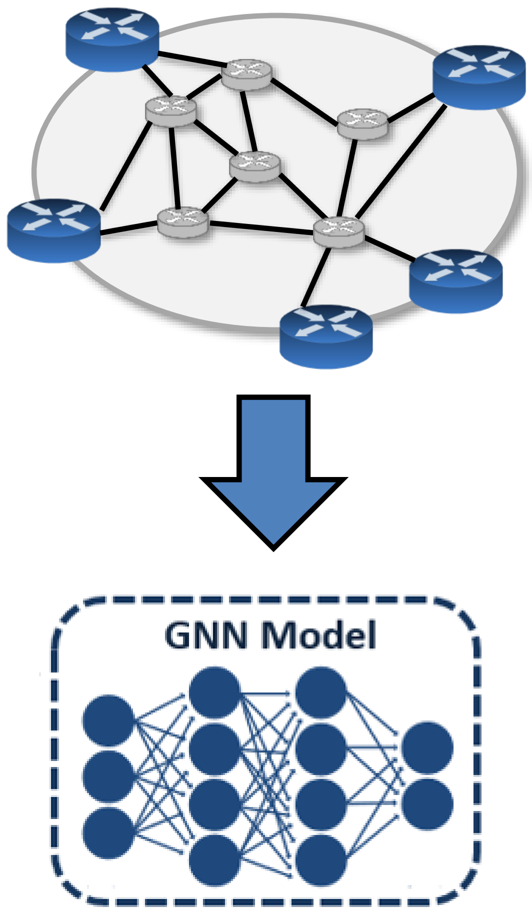
There was a temporary service disruption on Friday that affected some customer SLAs:

- Can we find a more robust configuration (e.g., routing, queue scheduling, link upgrade) that prevents this problem in the future?

Network Optimization and What-if Analysis

- What is the optimal configuration that guarantees the customer SLAs having minimum impact on best-effort traffic?
- How should we re-route traffic in case of a link failure and still satisfy SLAs? (**Traffic Engineering**)

Comparison against other network modeling alternatives

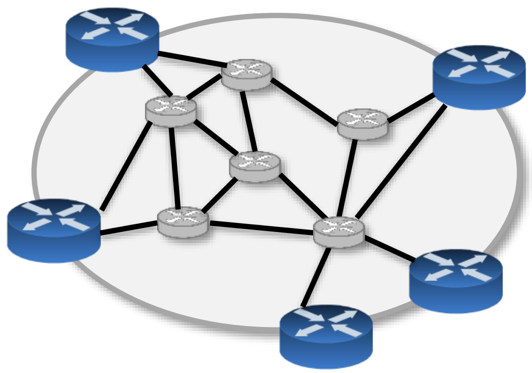


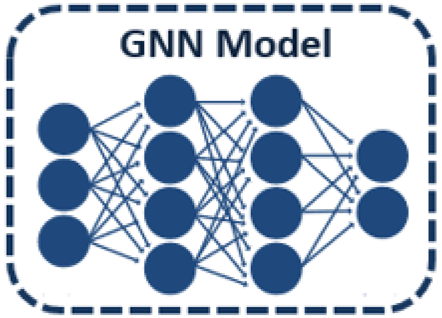
- Traditional alternatives to build digital twins:
  - ❌ **Network simulation** → Accurate, but computationally very expensive
    (They simulate each packet in the network!)
  - ❌ **Analytical solutions** (e.g., fluid models, queuing theory, network calculus)
    Inaccurate to model complex real-world networks [2]

[1] Rusek, K., Suárez-Varela, J., Mestres, A., Barlet-Ros, P. and Cabellos-Aparicio, A., 2019. Unveiling the potential of Graph Neural Networks for network modeling and optimization in SDN. *In ACM SOSR 2019.*

[2] Xu, Zhiyuan, et al. "Experience-driven networking: A deep reinforcement learning based approach." IEEE INFOCOM, 2018.

Comparison against other network modeling alternatives





- Traditional alternatives to build digital twins:
  - ❌ **Network simulation** → Accurate, but computationally very expensive (They simulate each packet in the network!)
  - ❌ **Analytical solutions** (e.g., fluid models, queuing theory, network calculus) Inaccurate to model complex real-world networks [2]

- Advantages of GNN with respect to state-of-the-art solutions:
  - ✅ **Fast** (low computational cost)
  - ✅ **Data-driven** → Higher accuracy (It can be trained with real data!)
  - ✅ **Deployability** → Train in a controlled testbed, apply to real-world networks

- **Graph Neural Networks is the only ML-based technique that is able to generalize to different networks [1]**

[1] Rusek, K., Suárez-Varela, J., Mestres, A., Barlet-Ros, P. and Cabellos-Aparicio, A., 2019. Unveiling the potential of Graph Neural Networks for network modeling and optimization in SDN. *In ACM SOSR 2019.*

[2] Xu, Zhiyuan, et al. "Experience-driven networking: A deep reinforcement learning based approach." IEEE INFOCOM, 2018.

Barcelona Neural Networking Center

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Departament d'Arquitectura de Computadors

Graph Neural Networking Challenge 2021

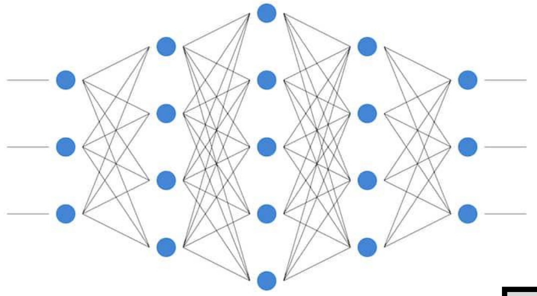
Creating a Scalable Network Digital Twin

https://bnn.upc.edu/challenge/gnnet2021/

How can we create commercial ML-based Network Digital Twins?

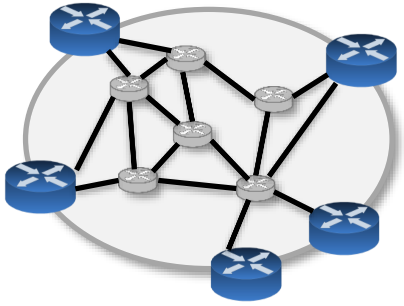
VENDOR’S NETWORKING LAB

ML model (e.g., neural network)



+

Controlled network testbed

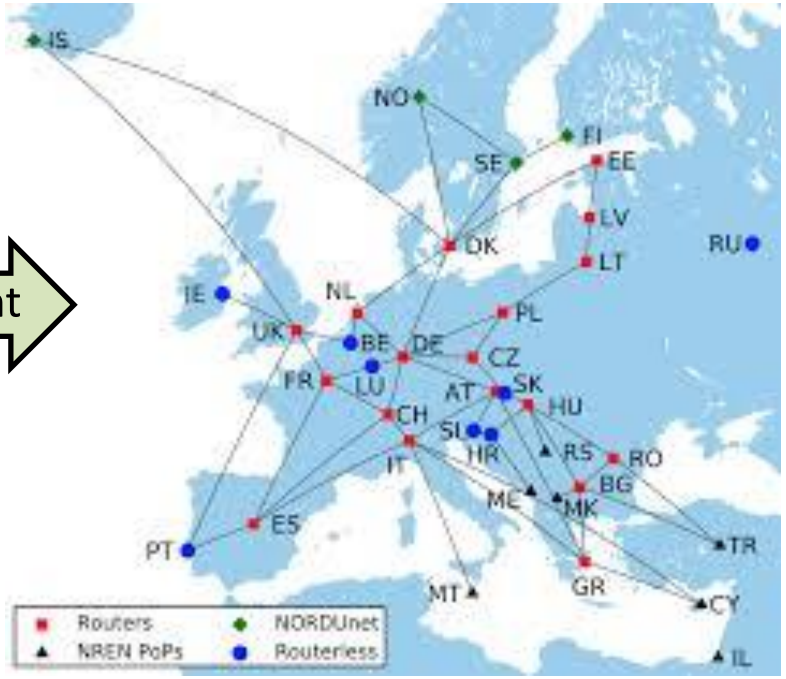


Training

DIGITAL TWIN

Deployment

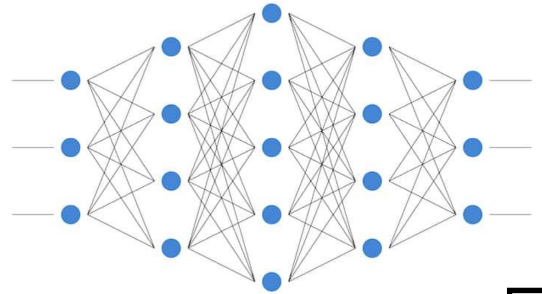
Customer’s network



How can we create commercial ML-based Network Digital Twins?

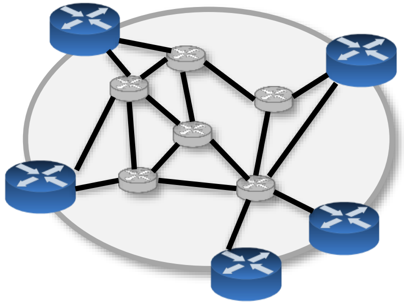
VENDOR'S NETWORKING LAB

ML model (e.g., neural network)



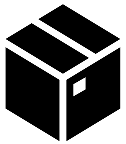
+

Controlled network testbed



Training

DIGITAL TWIN



Deployment

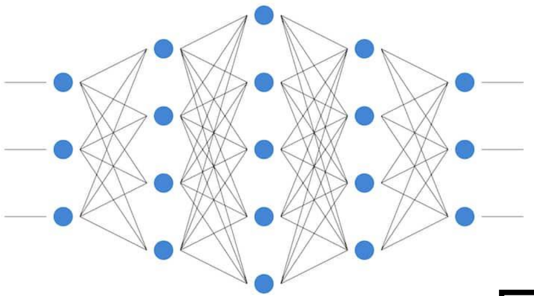
Customer's network

Training directly in the real network is not feasible, as it requires to cover cases that <u>may lead to service disruption</u> (e.g., generate link failures)

How can we create commercial ML-based Network Digital Twins?

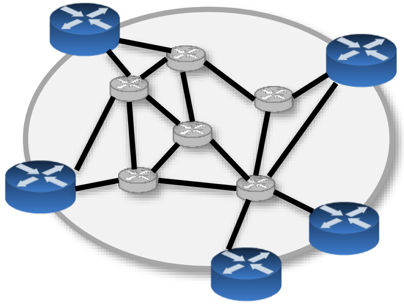
VENDOR'S NETWORKING LAB

ML model (e.g., neural network)



+

Controlled network testbed



Training

DIGITAL TWIN

Deployment

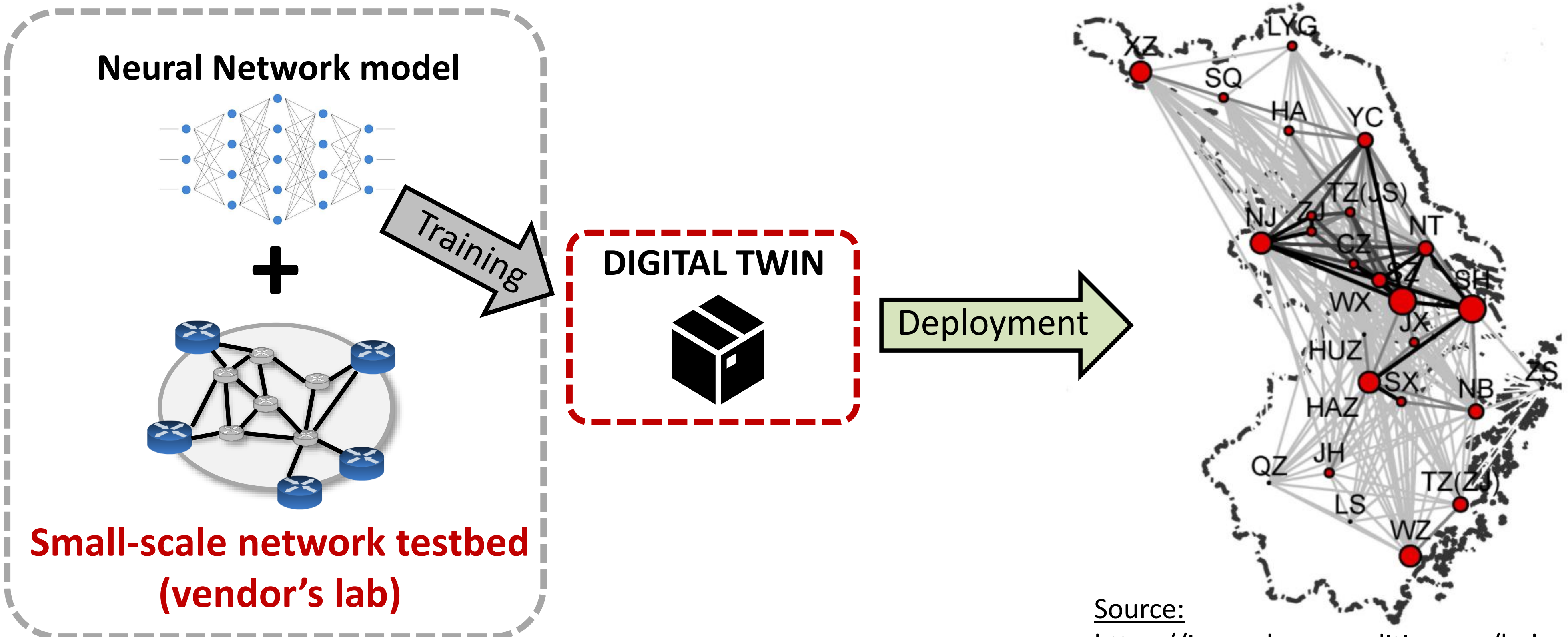
Need for ML models that can generalize to other networks not seen during training
(Graph Neural Networks)

Customer's network

Training directly in the real network is not feasible, as it requires to cover cases that <u>may lead to service disruption</u> (e.g., generate link failures)

Graph Neural Networking challenge 2021

- **Problem overview:**

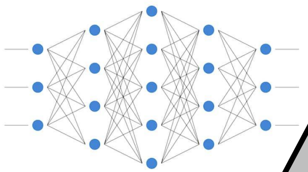


Source:
https://journals.openedition.org/belgeo/17087

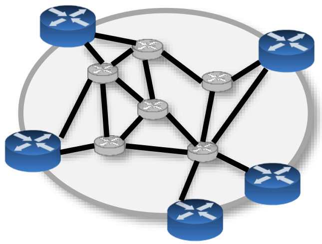
- **Problem overview:**

Creating a scalable Network Digital Twin

Neural Network model



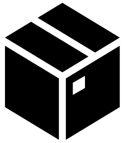
+



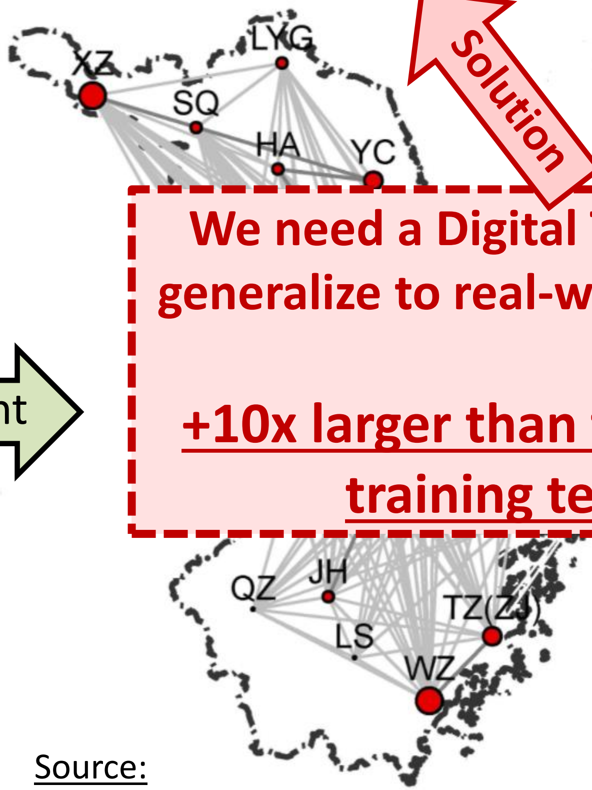
Small-scale network testbed (vendor's lab)

Training

DIGITAL TWIN



Deployment



Solution

We need a Digital Twin that can generalize to real-world networks!

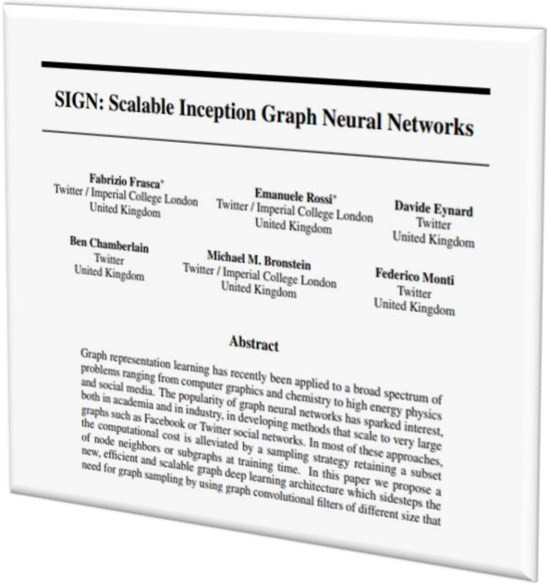
+10x larger than the vendor's training testbed

Source: https://journals.openedition.org/belgeo/17087

- **<u>Problem overview:</u>**

Creating a <u>scalable</u> Network Digital Twin

- **Building scalable GNN models is an open problem in the ML field!**
- There are **few recent proposals** tailored to specific problems (e.g., classification)

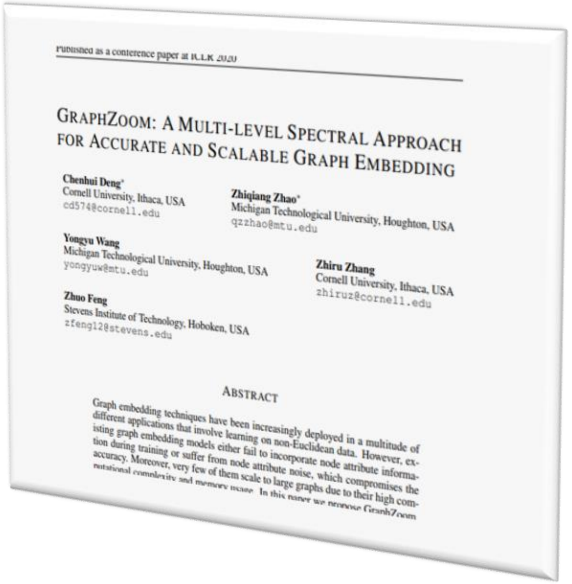


SIGN: Scalable Inception Graph Neural Networks

Fabrizio Frasca* Twitter / Imperial College London United Kingdom
Emanuele Rossi* Twitter / Imperial College London United Kingdom
Davide Eynard Twitter United Kingdom
Ben Chamberlain Twitter United Kingdom
Michael M. Bronstein Twitter / Imperial College London United Kingdom
Federico Monti Twitter United Kingdom

Abstract

Graph representation learning has recently been applied to a broad spectrum of problems ranging from computer graphics and chemistry to high energy physics and social media. The popularity of graph neural networks has sparked interest, both in academia and in industry, in developing methods that scale to very large graphs such as Facebook or Twitter social networks. In most of these approaches, the computational cost is alleviated by a sampling strategy retaining a subset of node neighbors or subgraphs at training time. In this paper we propose a new, efficient and scalable graph deep learning architecture which sidesteps the need for graph sampling by using graph convolutional filters of different size that



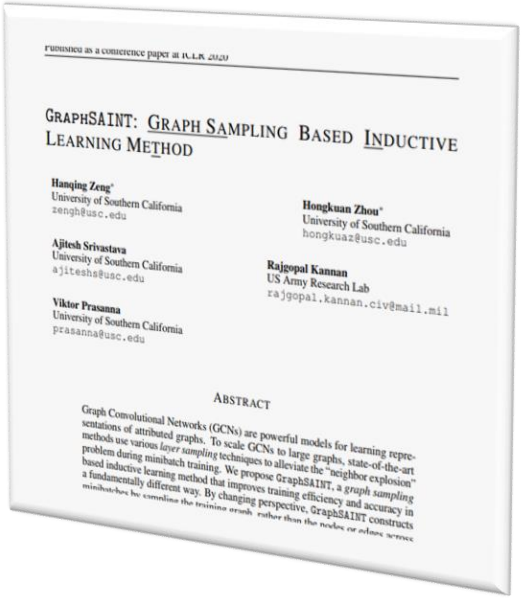
Published as a conference paper at ICLR 2020

GRAPHZOOM: A MULTI-LEVEL SPECTRAL APPROACH FOR ACCURATE AND SCALABLE GRAPH EMBEDDING

Chenhui Deng* Cornell University, Ithaca, USA cd574@cornell.edu
Zhiqiang Zhao* Michigan Technological University, Houghton, USA qzzhao@mtu.edu
Yongyu Wang Michigan Technological University, Houghton, USA yongyuw@mtu.edu
Zhiru Zhang Cornell University, Ithaca, USA zhiruz@cornell.edu
Zhuo Feng Stevens Institute of Technology, Hoboken, USA zfeng12@stevens.edu

ABSTRACT

Graph embedding techniques have been increasingly deployed in a multitude of different applications that involve learning on non-Euclidean data. However, existing graph embedding models either fail to incorporate node attribute information during training or suffer from node attribute noise, which compromises the accuracy. Moreover, very few of them scale to large graphs due to their high computational complexity and memory usage. In this paper we propose GraphZoom



Published as a conference paper at ICLR 2020

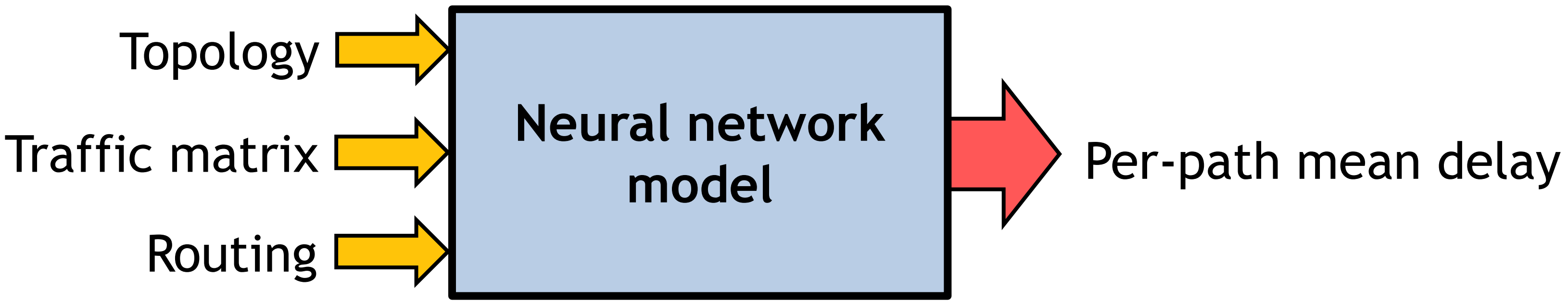
GRAPHSAINT: GRAPH SAMPLING BASED INDUCTIVE LEARNING METHOD

Hanqing Zeng* University of Southern California zengh@usc.edu
Hongkuan Zhou* University of Southern California hongkuaz@usc.edu
Ajitesh Srivastava University of Southern California ajiteshs@usc.edu
Rajgopal Kannan US Army Research Lab rajgopal.kannan.civ@mail.mil
Viktor Prasanna University of Southern California prasanna@usc.edu

ABSTRACT

Graph Convolutional Networks (GCNs) are powerful models for learning representations of attributed graphs. To scale GCNs to large graphs, state-of-the-art methods use various *layer sampling* techniques to alleviate the "neighbor explosion" problem during minibatch training. We propose GraphSAINT, a *graph sampling* based inductive learning method that improves training efficiency and accuracy in a fundamentally different way. By changing perspective, GraphSAINT constructs minibatches by sampling the training graph, rather than the nodes or edges across

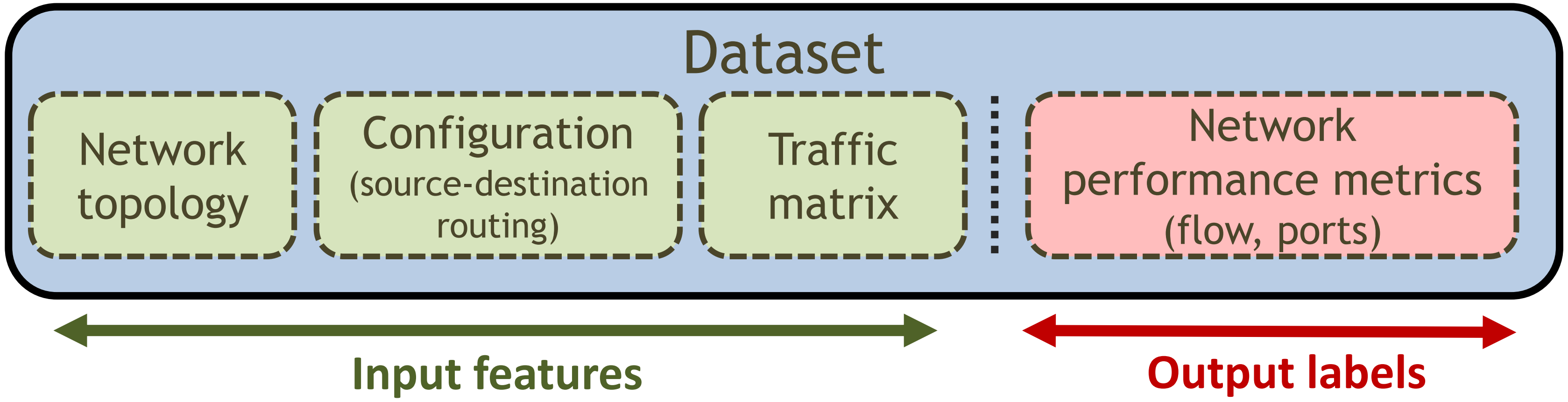
Objective: Build this box



- Input:
  - Network topology
  - Source-destination traffic matrix
  - Routing (source-destination paths)

- Output:
  - Mean per-packet delay on each source-destination flow

Datasets



- Data generated with OMNet++ (packet-accurate network simulator)
- Thousands of simulation samples with topologies, routing configurations, and traffic (covering wide range of congestion levels)

Datasets

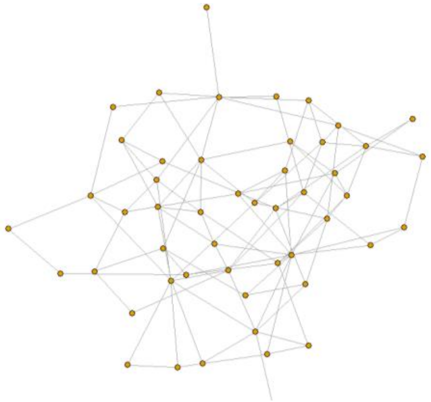
<u>Challenge objective</u> → Test the scalability properties of proposed solutions

Training dataset:

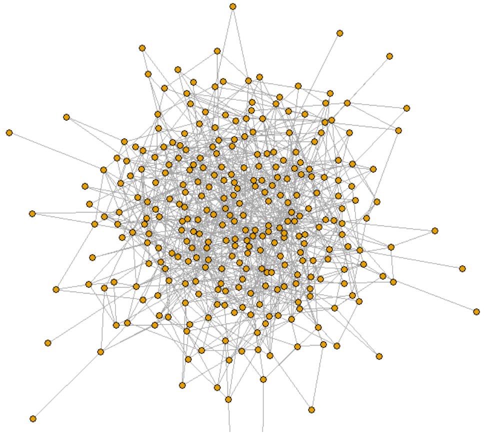
- Samples in a wide variety of networks from **25 to 50 nodes** (small)

Validation/Test datasets:

- Samples in networks from **51 to 300 nodes** (large)



Training (small networks)



Validation / Test (large networks)

Evaluation

- **Evaluation score →** MAPE (Mean Absolute Percentage Error)

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{\hat{y}_i - y_i}{y_i}\right|$$

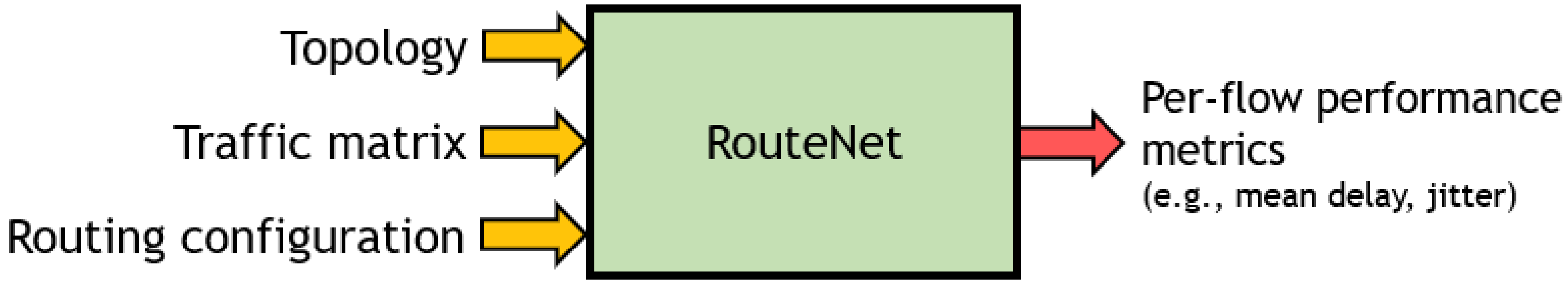
Lower is better!

- At the end of the challenge (Sep 15^{th}), we will release the **test dataset** (unlabeled)

- **The evaluation phase lasts 15 days**, and is made automatically in our evaluation platform

- **Participants will be able to see their score in real time**

Baseline model

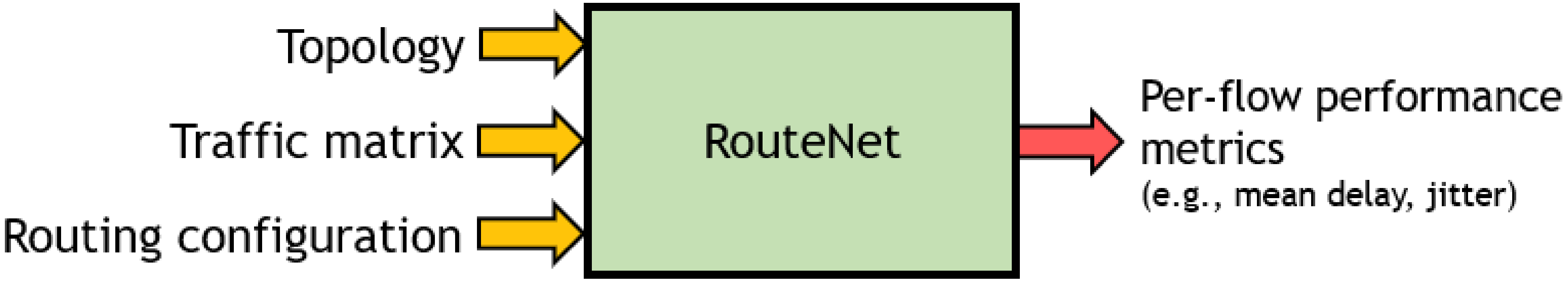


- RouteNet* learns the relations between topology, traffic, routing and how these elements affect the resulting network performance (e.g., delay)

- Generalizes to **unseen** topologies, routing configurations and traffic

*Rusek, K., Suárez-Varela, J., Mestres, A., Barlet-Ros, P. and Cabellos-Aparicio, A., 2019. Unveiling the potential of Graph Neural Networks for network modeling and optimization in SDN. *In ACM SOSR 2019*

Baseline model



Limitation:

RouteNet* is not designed to model networks (graphs) considerably larger than those seen during the training phase (MAPE > 300% in our experiments)

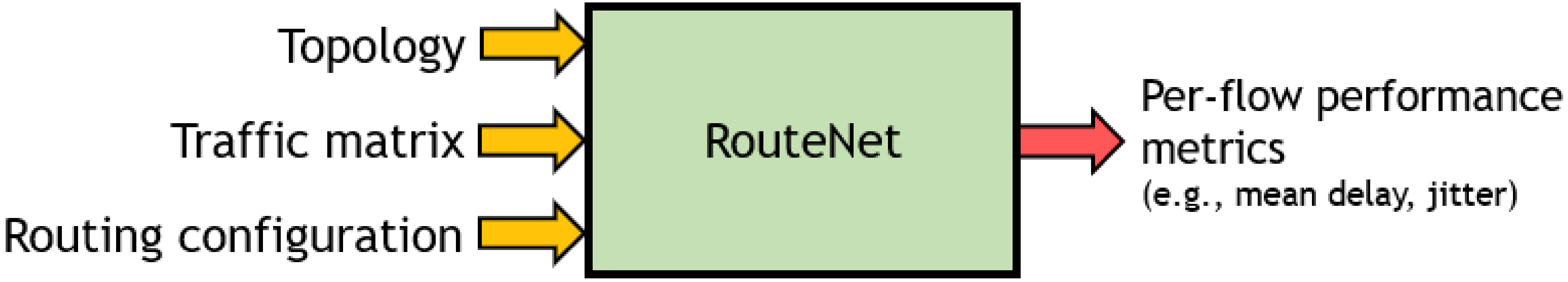
- We provide open source implementations in TensorFlow and IGNNITION

tensorflow.org ignnition.net

*Rusek, K., Suárez-Varela, J., Mestres, A., Barlet-Ros, P. and Cabellos-Aparicio, A., 2019. Unveiling the potential of Graph Neural Networks for network modeling and optimization in SDN. *In ACM SOSR 2019*

Guidelines for participants

- **Participants are encouraged to update RouteNet or design their own neural network architectures**
- We provide a tutorial on how to run RouteNet and modify fundamental parts of the code



Guidelines for participants

What means larger networks?

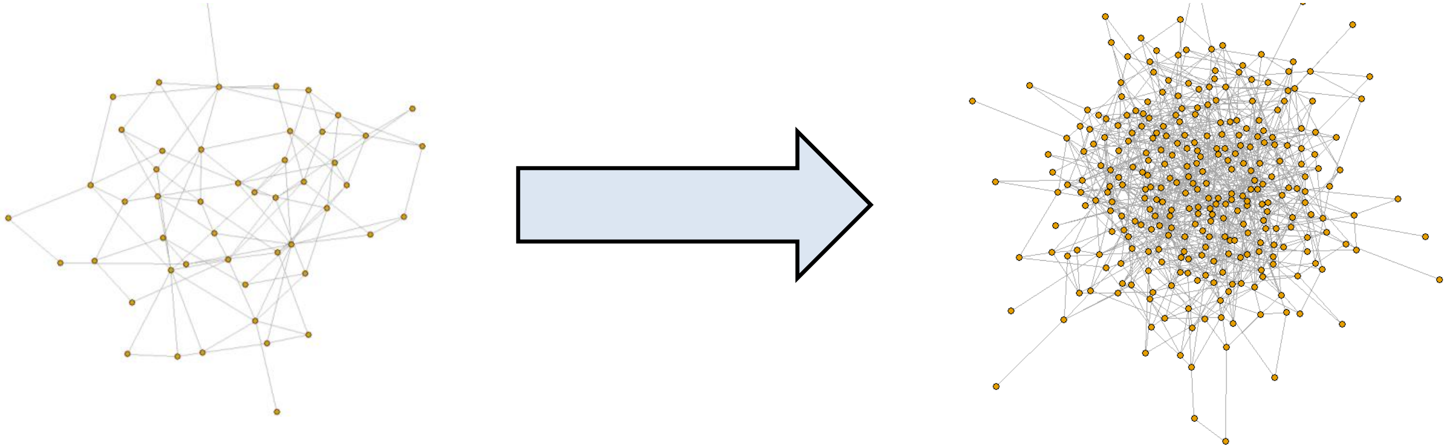
Two main features with respect to small networks (training):

1) Longer paths (higher network diameter)

2) Higher link capacities

We need a model that can effectively scale in these two features!

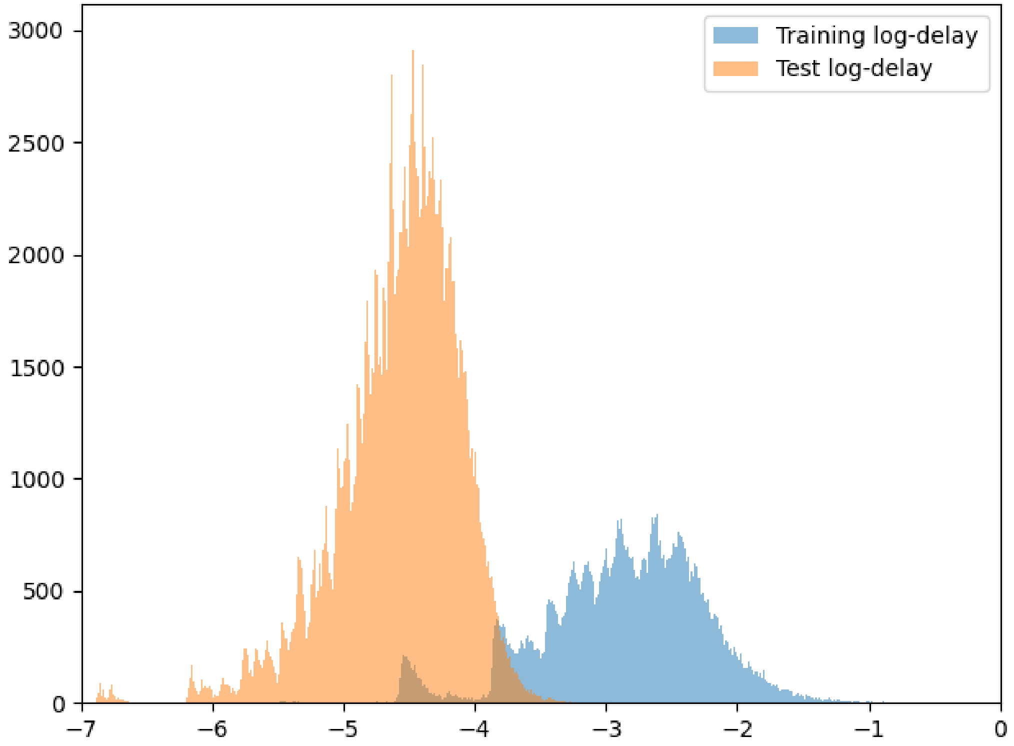
Training (small networks) **Validation / Test (large networks)**



What means larger networks?

Another obstacle:

- Output distributions of the delay (log scale)

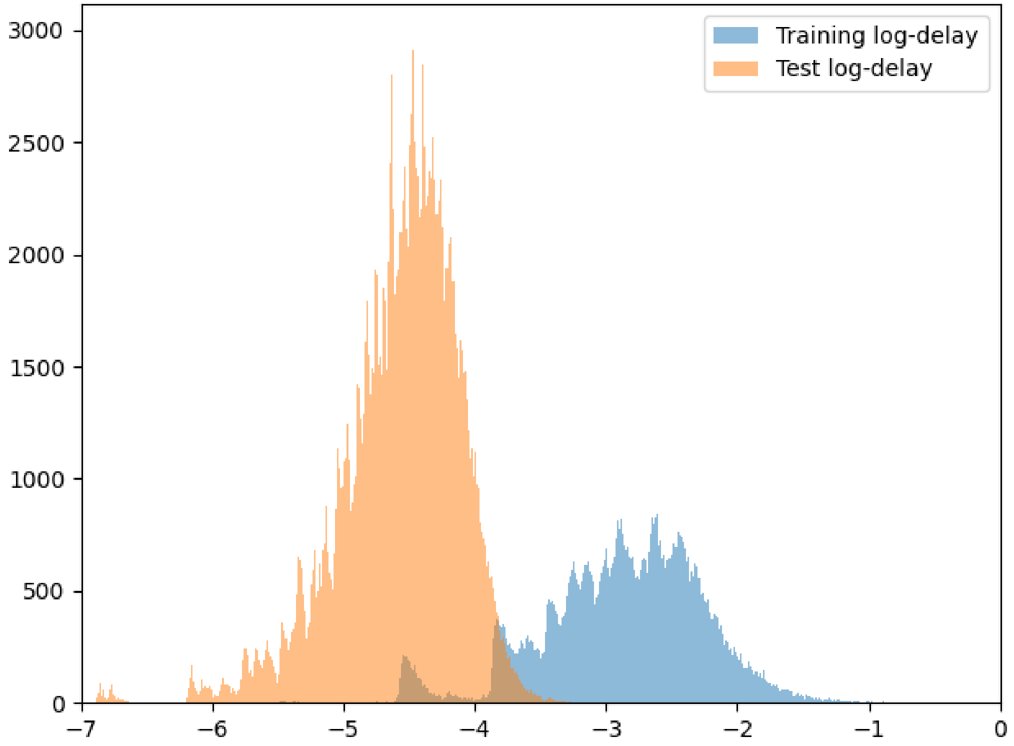


Producing out-of-distribution values with neural networks is an unsolved problem in the ML field!

What means larger networks?

Another obstacle:

- Output distributions of the delay (log scale)

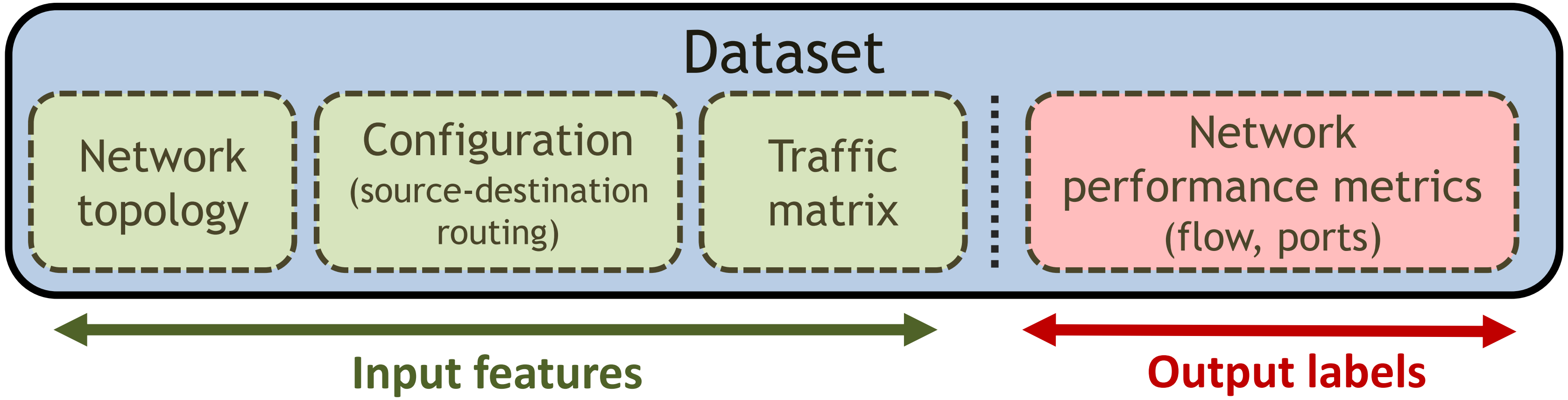


Producing out-of-distribution values with neural networks is an unsolved problem in the ML field!

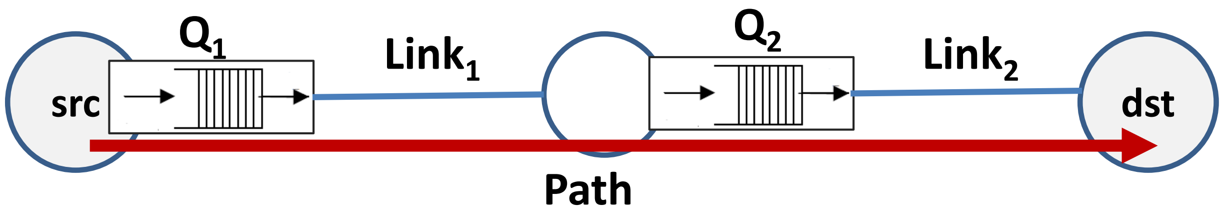
Can we transform the output of the NN to follow similar distributions in the training and test datasets?

What means larger networks?

Coming back to the datasets:



- **Port statistics** → Queue utilization ranges in [0,1] (both in training and test!)
- The NN model can **predict queue utilizations** and then infer path delays with a simple **post-processing**



$$\mathrm{Delay}_{L1} = \mathrm{Avg\_utilization}_{Q1} * \mathrm{Size}_{Q1} / \mathrm{Cap}_{L1}$$

$$\mathrm{Delay}_{Path} = \sum_{k=1}^{N\ links} Delay_{L_k}$$

Incentives for participants

- **Good opportunity to be introduced in the application of GNN for networking (Hot topic!)**

This is the only competition in the world on GNN applied to networks!

- **Large-scale event** → The previous edition had **more than 120 participants from 24 countries [1] [2]**

- **Top-3 teams will be publicly recognized** in the challenge website and will receive certificates

- **Possibility to publish a paper** co-authored with the challenge organizers

[1] https://aiforgood.itu.int/ai-ml-in-5g-challenge-2020
[2] https://bnn.upc.edu/challenge/gnnet2020

Incentives for participants

- **Good opportunity to be introduced in the application of GNN for networking (Hot topic!)**

This is the only competition in the world on GNN applied to networks!

- **Large-scale event** → The previous edition had **more than 120 participants from 24 countries [1] [2]**

- **Top-3 teams will be publicly recognized** in the challenge website and will receive certificates

- **Possibility to publish a paper** co-authored with the challenge organizers

- **Access to the global round of the ITU AI/ML in 5G challenge**:
  - Top candidates will be considered by the ITU judging committee
  - ITU Awards and presentation at the final conference (Dec 2021) [3]

[1] https://aiforgood.itu.int/ai-ml-in-5g-challenge-2020
[2] https://bnn.upc.edu/challenge/gnnet2020
[3] https://aiforgood.itu.int/events/itu-ai-ml-in-5g-grand-challenge-finale-final-conference/

Quick summary

Organized as part of the ITU AI/ML in 5G Challenge

- Special thanks to ITU for making this possible!

Website: https://aiforgood.itu.int/ai-ml-in-5g-challenge/

Target audience:

- Networking community
- ML community (Scalability in GNN is an relevant open challenge!)

Main resources:

- RouteNet*: Baseline model including a tutorial (open implementation in TensorFlow and IGNNITION)
- API to easily read and process the datasets
- Mailing list for Q&A from participants

*K. Rusek, J. Suárez-Varela, A. Mestres, P. Barlet-Ros, A. Cabellos-Aparicio, “Unveiling the potential of Graph Neural Networks for network modeling and optimization in SDN,” In Proc. of ACM SOSR, 2019.

Organizing team

José Suarez-Varela

Albert López

Miquel Ferriol

Guillermo Bernárdez

Paul almasan

Krzysztof Rusek

Barcelona
Neural
Networking
Center

Prof. Pere Barlet-Ros

Prof. Albert Cabellos

Timeline

Check all the details at:

https://bnn.upc.edu/challenge/gnnet2021

- **Challenge duration:** May 20th-Nov 20th 2021
- **Open registration for participants!**
- **Score-based evaluation phase:** Sep 16th-Sep 30th
- **Final ranking and official announcement of top-3 teams:** Oct 31st

Subscribe to our mailing list [link]

Website: [here]
Slack channel: [here]