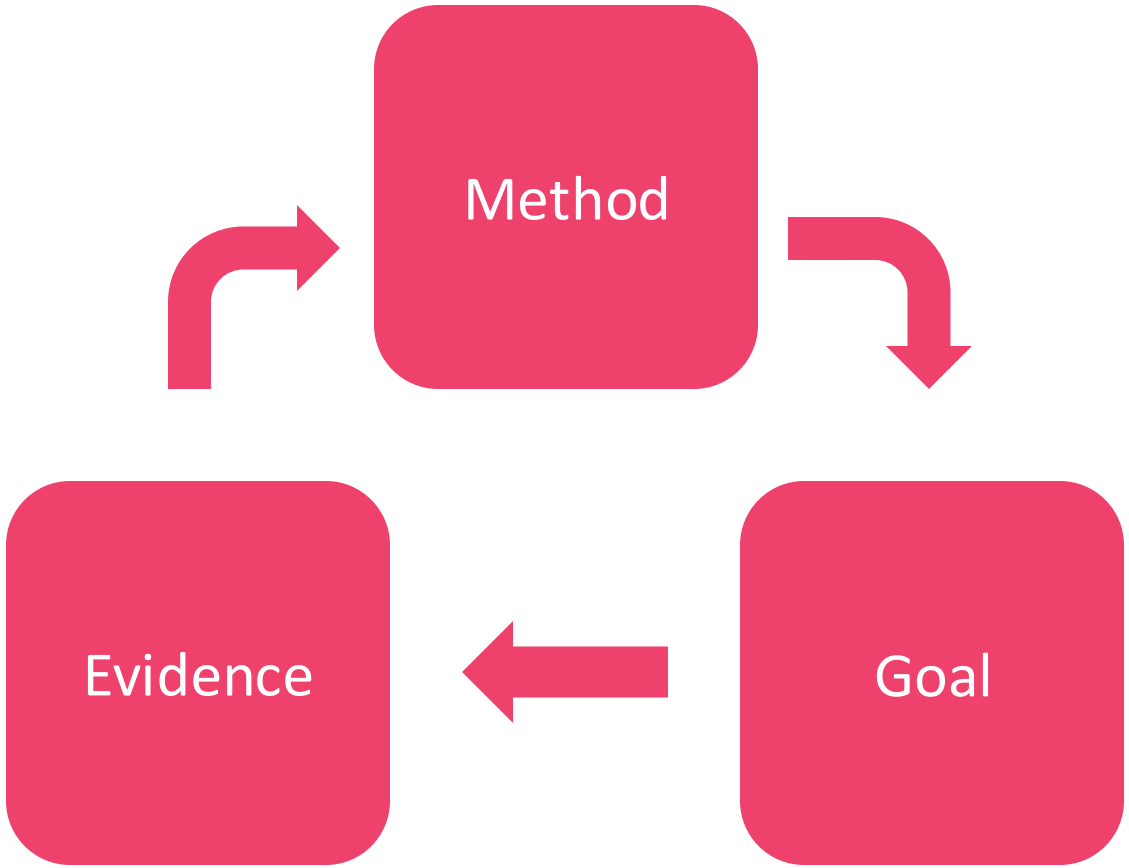
Using *Graph Neural Networks* to classify traffic types in GÉANT's backbone network

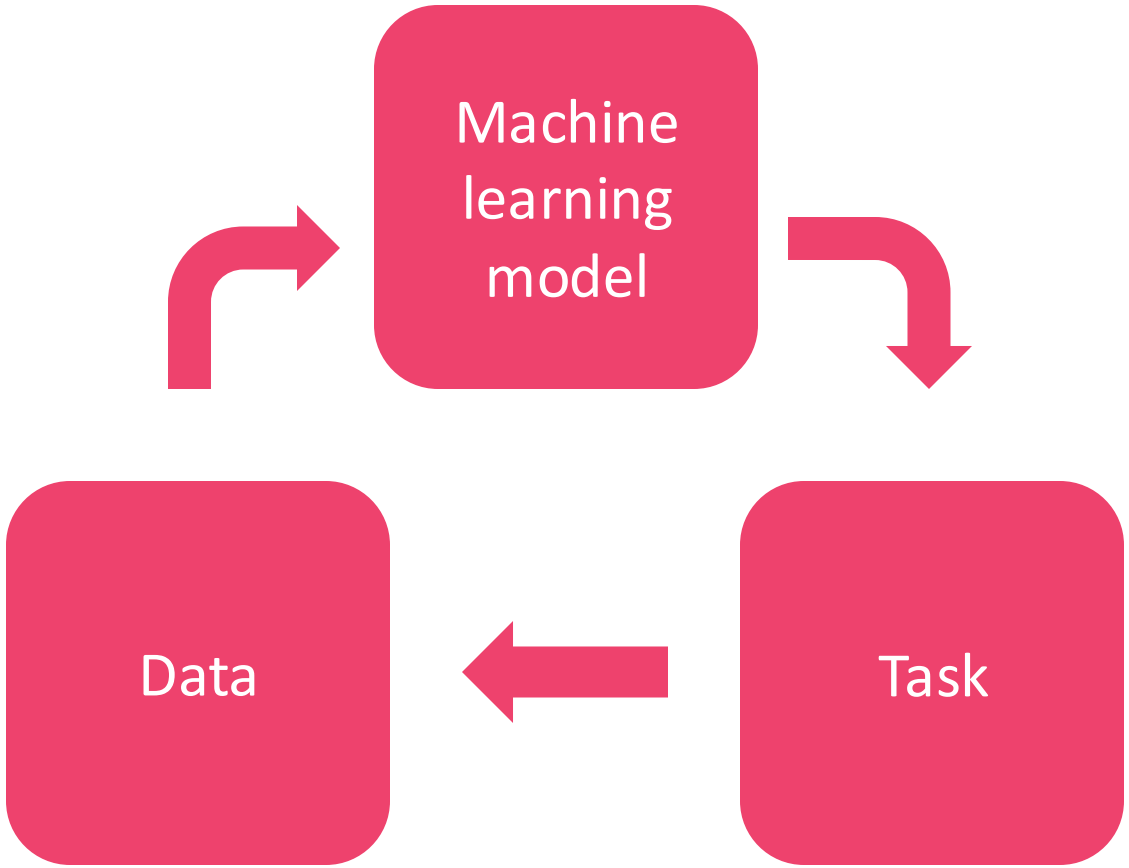
Maarten Meijer
MSc graduate intern AI
University of Twente
maarten.meijer@geant.org

SIG-AI, Prague
7 April 2025

Data Science flow

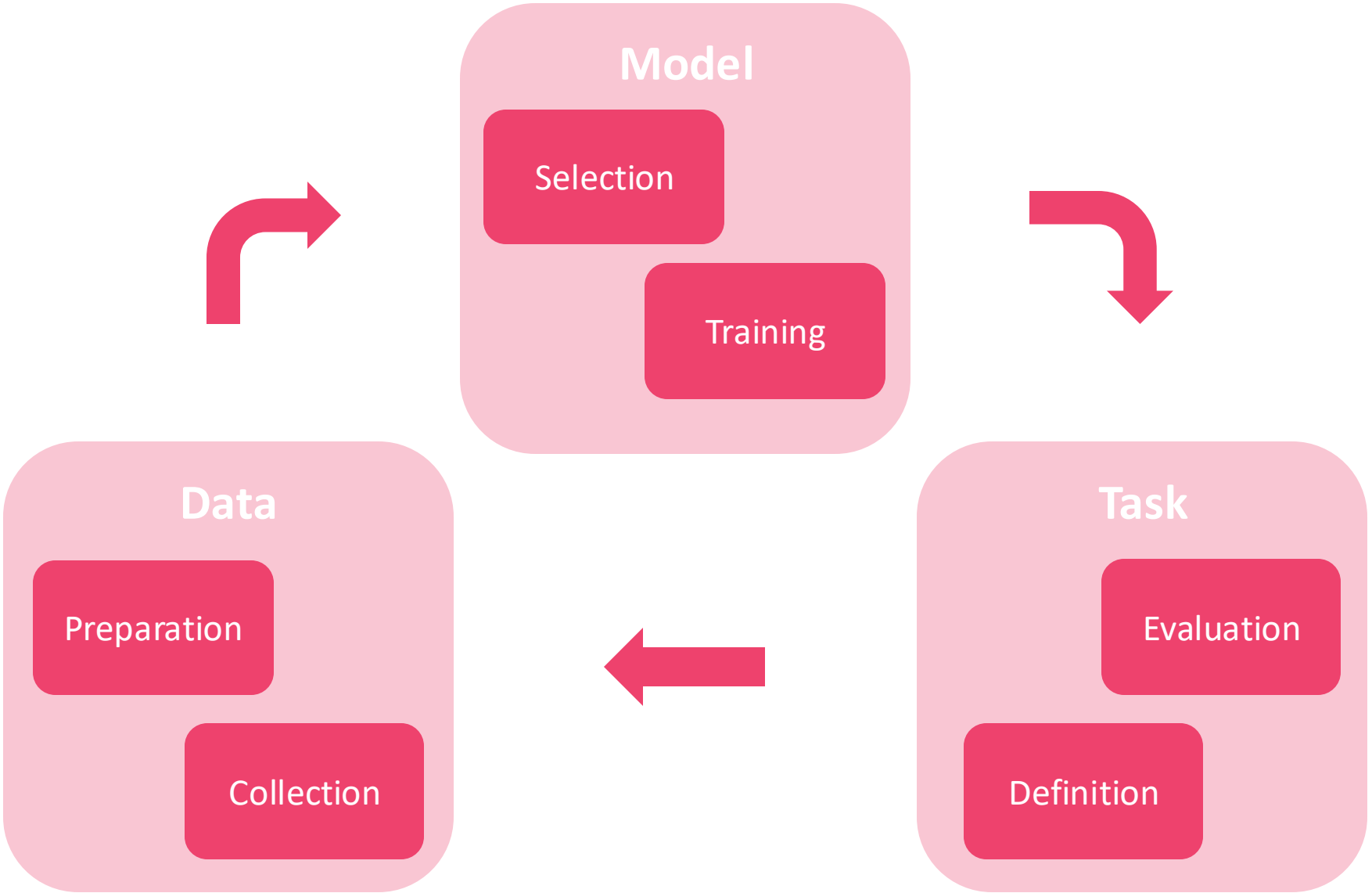


Data Science flow



Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a Standard Process Model for Data Mining.

Data Science flow

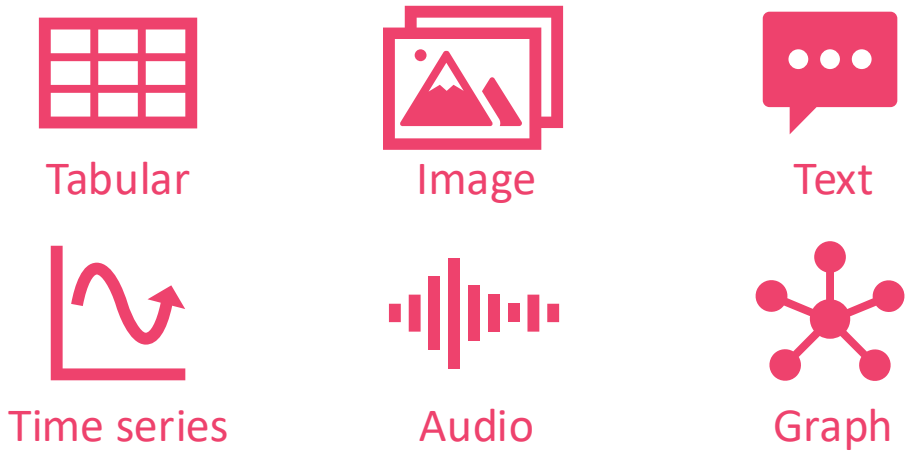


Task *(in computer networks)*

- Anomaly detection (and classification)
- Estimate network congestion
- Forecast network traffic
- Learn representation of network traffic
- Assess network performance
- Node congestion
- Traffic generation

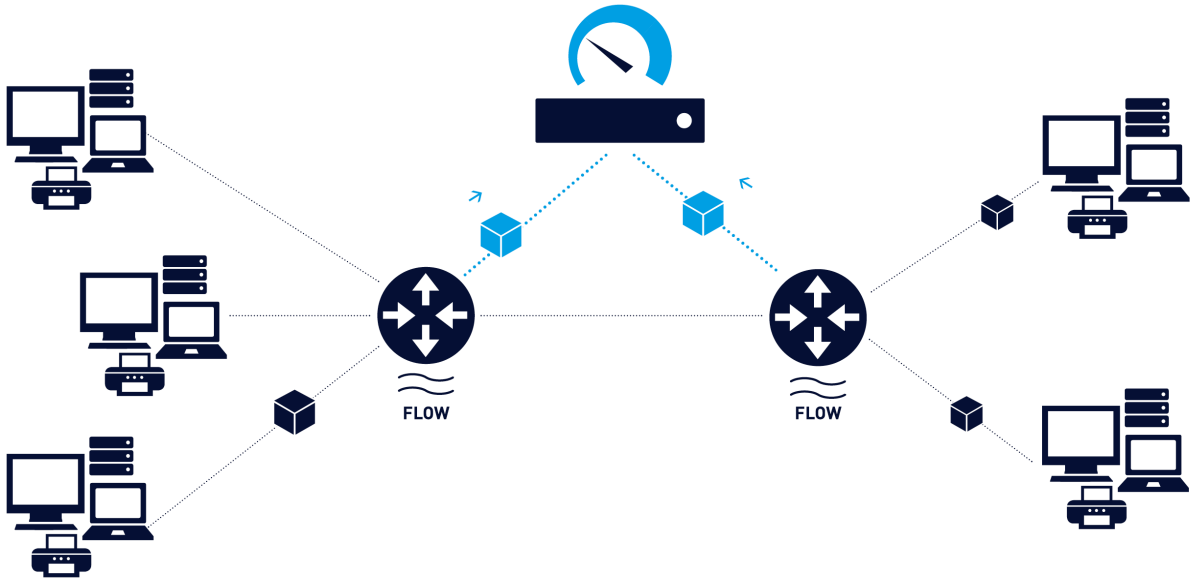
Data

- Some common data types



NetFlow Data

- Grouping of unidirectional stream of packets
- Configurable
- Sampling
- NetFlow collector



Juniper IPFIX Fields
IPv4 Source Address
IPv4 Destination Address
IPv4 ToS
IPv4 Protocol
L4 Source Port
L4 Destination Port
ICMP Type and Code
Input Interface
⋮
TCP Flags
Minimum TTL
Maximum TTL
Number of Flow Bytes
Number of Flow Packets
Time the Flow Started
Time the Flow Ended

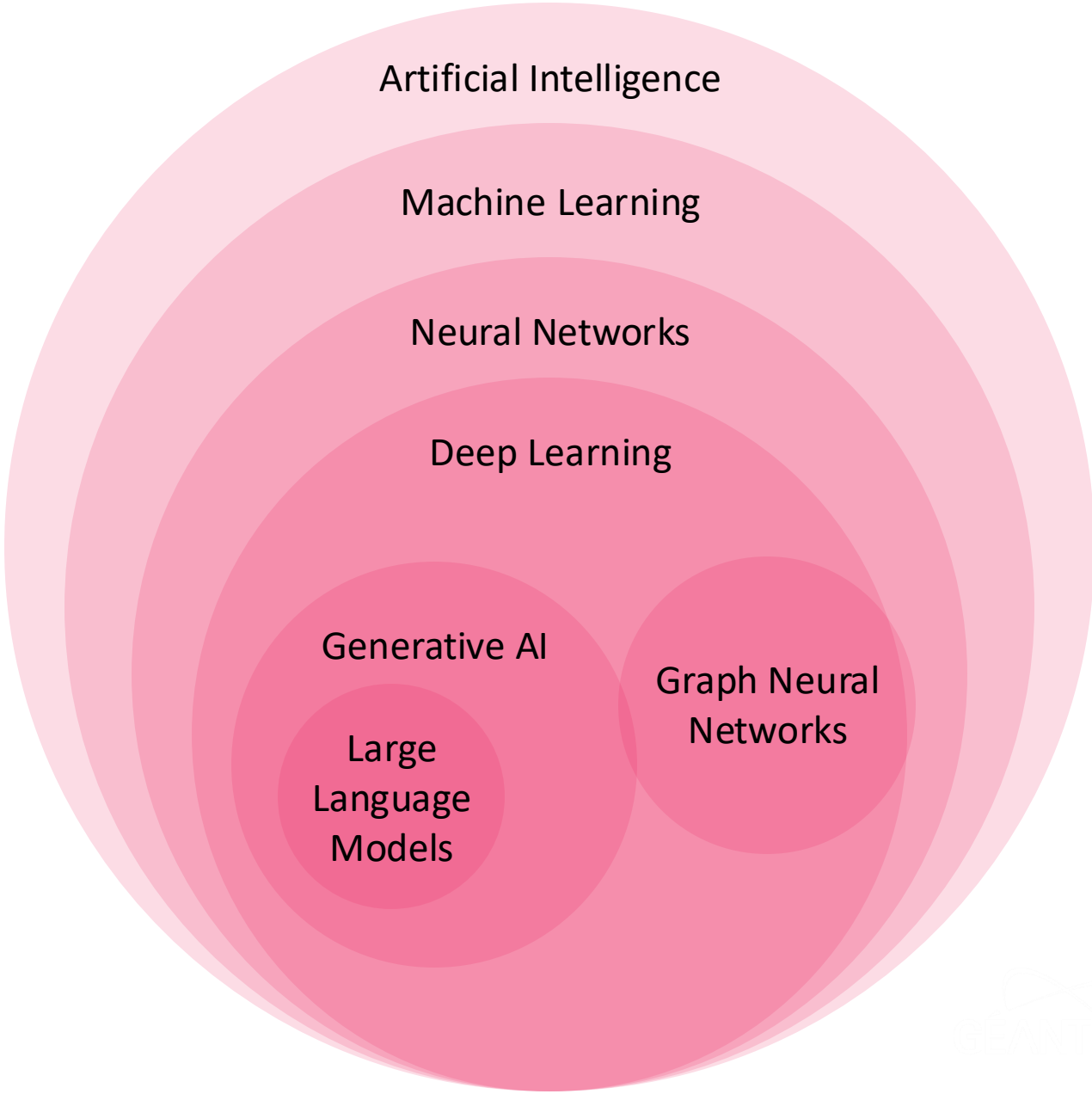
NetFlow Data

- Target labels
  - Benign, DDoS, Reconnaissance, Injection, DoS, Brute Force, Password, XSS, Infiltration, etc.
- Public datasets
  - UNSW-NB15[1], BoT-IoT[2], ToN-IoT[3]
  - UNSW Canberra Cyber Range Lab

[1] *Moustafa, N., & Slay, J. (2015). UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set).*
[2] *Koroniotis, N., Moustafa, N., Sitnikova, E., & Turnbull, B. (2018). Towards the development of realistic botnet dataset in the internet of things for network forensic analytics: Bot-IoT dataset.*
[3] *Moustafa, N. (2021). A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets.*

Model

- Connect data and task
- Supervised or unsupervised?
- Many options...



Caville, E., Lo, W. W., Layeghy, S., & Portmann, M. (2022). Anomal-E: A self-supervised network intrusion detection system based on graph neural networks.

Paper

Anomal-E: A Self-Supervised Network Intrusion Detection System based on Graph Neural Networks

Evan Caville[a,2,*], Wai Weng Lo[a,2,*], Siamak Layeghy[a], Marius Portmann[a]

[a]*School of ITEE, The University of Queensland, Brisbane, Australia*

Abstract

This paper investigates graph neural networks (GNNs) applied for self-supervised intrusion and anomaly detection in computer networks. GNNs are a deep learning approach for graph-based data that incorporate graph structures into learning to generalise graph representations and output embeddings. As traffic flows in computer networks naturally exhibit a graph structure, GNNs are a suitable fit in this context. The majority of current implementations of GNN-based network intrusion detection systems (NIDSs) rely on labelled network traffic. This limits the volume and structure of input traffic and restricts the NIDSs' potential to adapt to unseen attacks. These systems also rely on the use of node features, which may reduce the detection accuracy of these systems, as important edge (packet-level) information is not leveraged. To overcome these restrictions, we present Anomal-E, a GNN approach to intrusion and anomaly detection that leverages edge features and a graph topological structure in a self-supervised manner. This approach is, to the best of our knowledge, the first successful and practical approach to network intrusion detection that utilises network flows in a self-supervised, edge-leveraging GNN. Experimental results on two modern benchmark NIDS datasets display a significant improvement when using Anomal-E compared to raw features and other baseline algorithms. This additionally posits the potential Anomal-E has for intrusion detection on real-world network traffic.

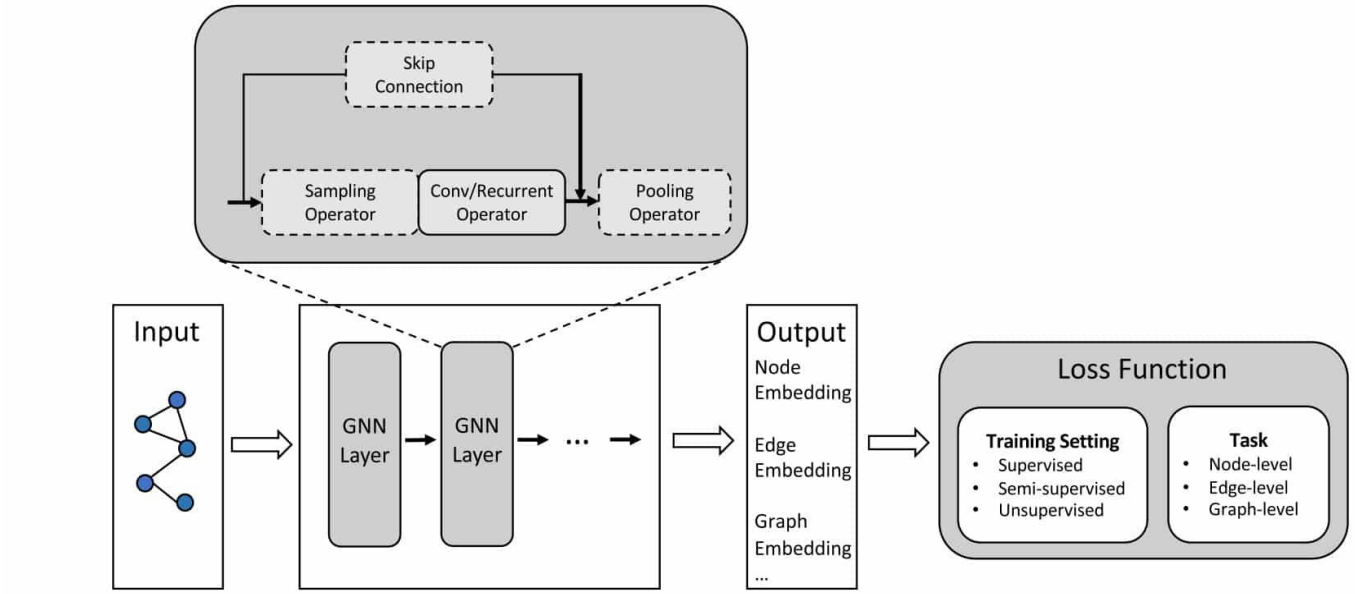
Keywords: graph neural network; network intrusion detection system; self supervised; graph representation learning; anomaly detection

1. Introduction

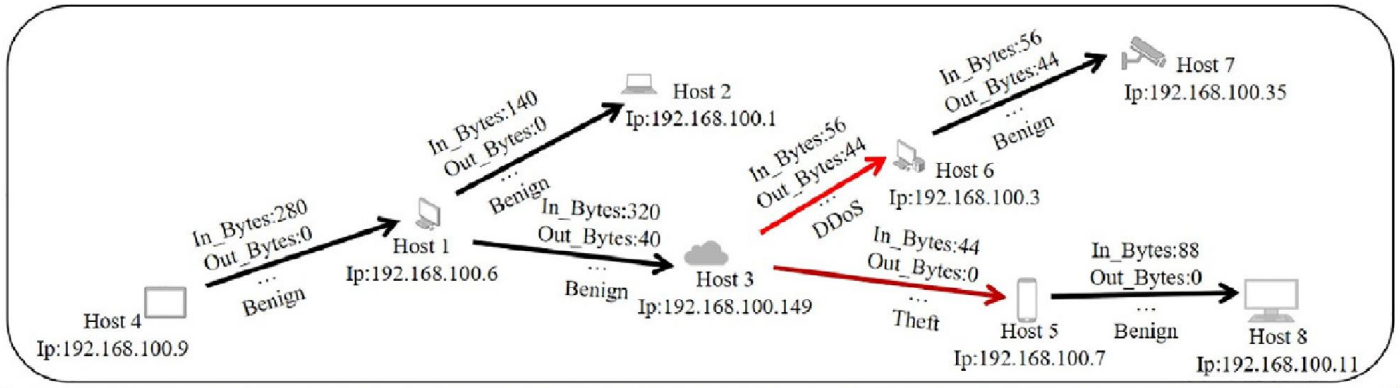
The increasing frequency and complexity of attacks on computer networks continues to threaten the security of information within computer systems. To combat this, enterprises factor in network intrusion detection. Due to this, we strongly argue for the inclusion of network flow topological patterns in NIDS development. This inclusion can be leveraged to detect sophisticated attacks, such as advanced persistent threat (APT)

Model

- Graph Neural Networks (GNNs)
- Connect data and task
  - Construct graph for edge classification
  - Self-supervised learning

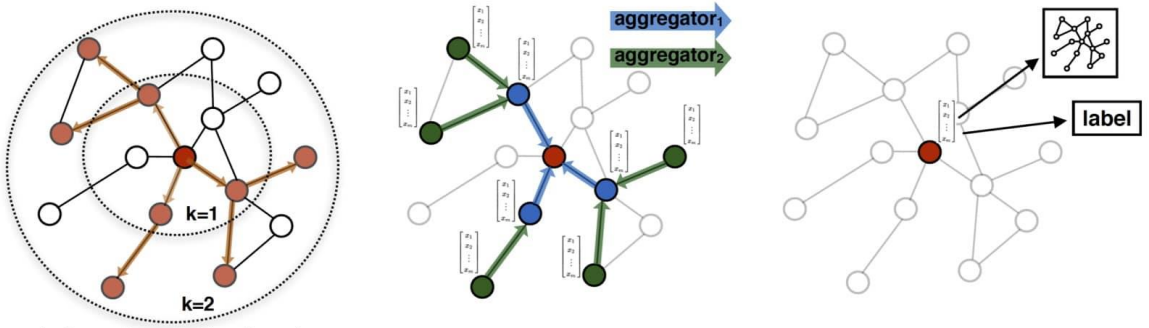


The general design pipeline for a GNN model.

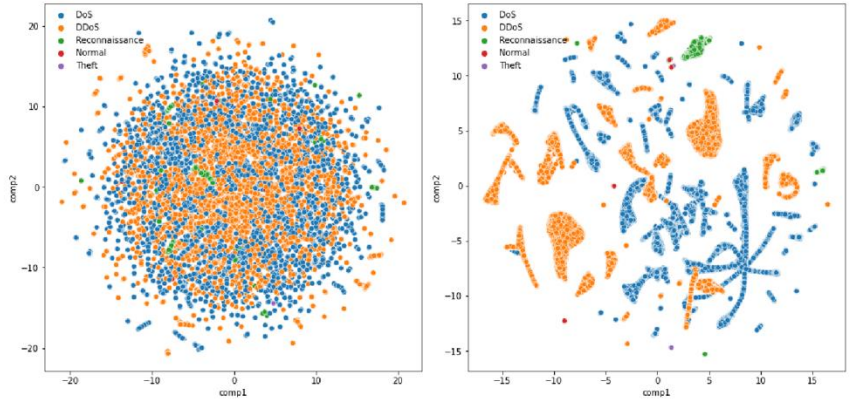


Model

- Experiments on GNNs
  - E-GraphSAGE[1], Anomal-E[2]
  - Outlier detection and clustering (unsupervised)



1. Sample neighborhood
2. Aggregate feature information from neighbors
3. Predict graph context and label using aggregated information



(a) raw data **(b)** edge embedded data

Fig. 4: Visualisation of dimensionality reduction a) Sample of BoT-IoT raw validation data, b)Sample of edge embeddings generated by E-GraphSAGE (Multiclass).

Table 6: NF-CSE-CIC-IDS2018-v2 results (4% contamination).

	Raw Features			Embeddings		
	Acc	Macro F1	DR	Acc	Macro F1	DR
PCA	85.91%	73.76%	74.71%	97.11%	92.57%	79.16%
IF	86.1%	74.09%	75.39%	89.79%	81.11%	91.84%
CBLOF	94.61%	86.18%	69.16%	97.80%	94.38%	82.67%
HBOS	88.81%	78.82%	84.22%	96.86%	91.89%	77.79%

[1] *Lo, W. W., Layeghy, S., Sarhan, M., Gallagher, M., & Portmann, M. (2022, April 25). E-GraphSAGE: A graph neural network based intrusion detection system for IoT.*
[2] *Caville, E., Lo, W. W., Layeghy, S., & Portmann, M. (2022). Anomal-E: A self-supervised network intrusion detection system based on graph neural networks.*

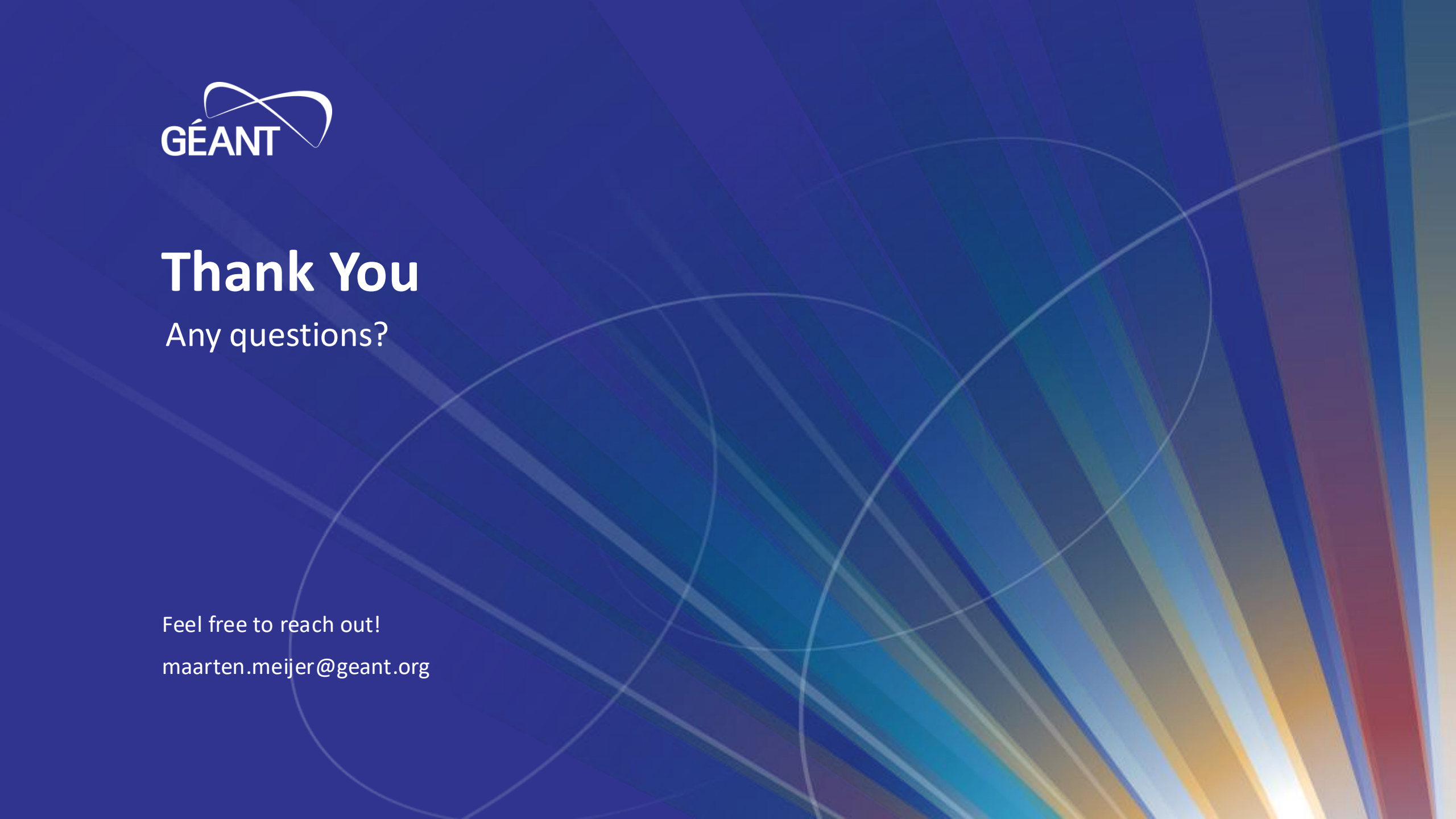
Experiment

- NVIDIA A16 GPU (8x16GB VRAM), 72 CPU cores, 256GB memory
- 1 month of flow data from Amsterdam router

Process	Time taken
Loading and preprocessing 230,000 flows	12s
Constructing graph	60s
Training Graph Neural Network	8m 54s (converging at 1700 train iterations)
Running model on test data	Instant!

Future steps

- Analyse specific flows in-depth
  - *What types of network traffic could exist?*
- Improve data
  - Higher quantity
  - Labeling (some) data
  - *Any other data to add?*
- Improve model
  - Use AS numbers as graph nodes
  - Research into GNNs
  - Add autoencoder instead of classical ML outlier detections
  - *Any NREN-specific research?*



Thank You

Any questions?

Feel free to reach out!

maarten.meijer@geant.org