Demystifying Network Foundation Models

Sylee (Roman) Beltiukov[1], Satyandra Guthula[1], Wenbo Guo[1], Walter Willinger[2], Arpit Gupta[1]

[1]University of California, Santa Barbara; [2]NIKSUN Inc.

Repo & Data & Additional Links

The Generalizability Paradox

Network Foundation Models (NFMs) promise **reusable, general-purpose representations** of network traffic, but in practice they **rarely transfer** across datasets, networks, or deployments.

In networking today, true **generalizability is still not achieved**, largely because our current evaluations **do not actually measure it!**

What's wrong with current evaluations?

A typical evaluation *before* and *after* finding all the problems in the benchmarking data looks like this:

	CIC-IDS-2017 *(Heartbleed)*[1]		**Crossmarket** *(Acc@10)*[2]	
	Original	Fixed	Original	Fixed
ET-BERT	99.99 ± 0.01	0.0 ± 0.0	99.82 ± 0.03	35.62 ± 0.39
YaTC	99.99 ± 0.01	0.01±0.01	99.69 ± 0.03	58.13 ± 0.89

[1]*AI/ML and Network Security: The Emperor has no Clothes, Jacobs et al, CCS '22;* [2]*netFound: Foundation Model for Network Security, Guthula et al, arXiv:2310.17025*

Reasons:
- Benchmarking datasets have ***shortcuts***, ***out-of-distribution issues***, and ***spurious correlations***.
- Benchmarking concentrates on **downstream tasks only**, mixing pretraining and fine-tuning, so we can't tell whether the pretrained model itself is broken.

Our Research Question:
Can we design an evaluation framework that:
1. Accounts for network context and application behaviour;
2. Avoids fine-tuning and downstream dataset shortcuts;
3. Reveals what pretrained models actually learn about networking?

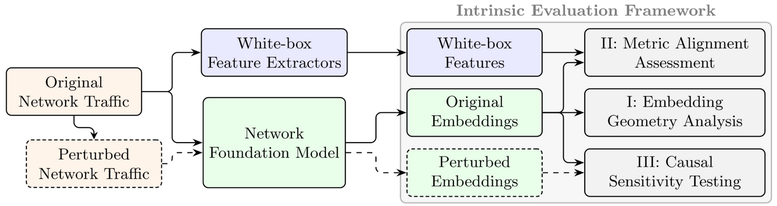
What we study: NFMs & datasets

All open-weights Network Foundation Models*:
- YaTC (Zhao et al, AAAI'23)
- ET-BERT (Lin et al, WWW'22)
- netFound (Guthula et al, arXiv:2310.17025)
- NetMamba (Wang et al, arXiv:2405.11449v4)

**Spring 2025*

Datasets:
- Endogenous (controlled) datasets:
  - Android Crossmarket
  - CIC-IDS-2017
  - CIC-APT-IIoT24
- Exogenous (real-world) datasets:
  - CAIDA
  - MAWI

Intrinsic Evaluation Framework (IEF)



IEF uses **embeddings** from a pretrained frozen NFM to understand:
- how **geometrically** efficient they are;
- what **information** they contain;
- and how **sensitive** they are to the **network context**.

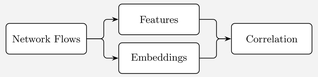
Part I: Embedding Geometry

- *Goal:* understand how efficiently the NFM **utilizes the embedding space**.
- *Technique:* measure embeddings' **anisotropy score** via random embedding pairs selection inside datasets and investigate both anisotropy score and mean dimensional contribution (MCC).

$$\mathcal{A} = \mathbb{E}_{i \neq j}[\cos(h_i, h_j)] \approx \frac{1}{|S|} \sum_{(i,j) \in S} \cos(h_i, h_j)$$

Part II: Metric Alignment

- *Goal:* evaluate presence of **well-known** statistical **network features** within NFMs embeddings.
- *Technique:* calculate aggregated white-box network features using **CICFlowMeter** and measure **Centered Kernel Alignment (CKA)** similarity index between them and embeddings.



Part III: Causal Sensitivity Testing

- *Goal:* observe NFMs sensitivity to **protocol-relevant** and **exogenous** network context changes.
- *Techniques:*
  a. Apply **perturbations** to different **header fields** of network packets and observe cosine similarity between the perturbed and original embeddings.
  b. Generate **realistic** network traffic with different **high-level contexts** (Congestion Control, Active Queue Management policy, and Cross Traffic) and observe cosine similarity changes and linear probing F_1 score.

Key Findings

Findings I: Embedding Geometry

- The analysis reveals different failure modes:

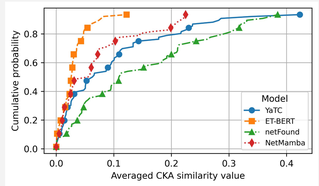
	Avg cos sim	*Top MCC*	*Comment*
NetMamba	**0.96**	0.02	Highly clustered embeddings
YaTC	0.86	**0.23**	Single dimension is overused

- Models show **worse** anisotropy on **real-world traffic** compared to exogenous datasets.
- You can even **improve F_1 score** on some downstream tasks without additional training by just **whitening** the embeddings:

	Crossmarket	*CIC-IDS-2017*	*CIC-APT-IIoT24*
NetMamba	+0.35 ± 0.02	+0.11 ± 0.27	+0.03 ± 0.02

Findings II: Metric Alignment

- **Architectural choices** and **input modality** largely determine how well NFMs encode white-box metrics.
  - *Temporal-spatial features dominate alignment signal;*
  - *Payload-only models (ET-BERT) struggle compared to feature-diverse models (netFound, YaTC, NetMamba).*



- Models again **fail** on endogenous data:

Average Feature Similarity	
Real-world datasets	Controlled environments
0.044	0.111

Findings III: Causal Sensitivity Testing

- Models *sometimes* can capture **high-level concepts** not presented in the input traffic:

	YaTC	**ET-BERT**	**netFound**	**NetMamba**
Linear probing	*F_1 score*	*F_1 score*	*F_1 score*	*F_1 score*
Congestion Control	0.48	0.41	**0.60**	0.29
AQM	0.51	0.68	**0.93**	0.70
Crosstraffic	0.24	0.43	**0.78**	0.33

- But they also pay too much attention to the **encrypted** user traffic **payload**, memorizing datasets!

	YaTC		**ET-BERT**		**netFound**		**NetMamba**	
	% tok	*cos*	% tok	*cos*	% tok	*cos*	% tok	*cos*
Payload	75%	0.18	100%	0.48	33%	0.99	75%	0.62

Takeaways
- NFMs show severe **representation pathologies** despite high benchmark scores;
- Downstream-only evaluation hides these issues, so **intrinsic geometry, metric,** and **causal tests** are necessary;
- These insights give concrete directions to **improve NFMs** and their training.