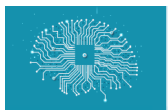
Formal Verification for Deep Learning-based Mobile Network Traffic Prediction

Thanh Le, Takeshi Matsumura

February 24, 2026

NICT WIRELESS NETWORKS RESEARCH CENTER
WIRELESS SYSTEMS LABORATORY



ICAIIC 2026

The 8th International Conference on Artificial Intelligence in Information and Communication

February 24 (Tue.) ~ 27 (Fri.), 2026, Tokyo University of Science, Tokyo, Japan & Virtual Conference

Agenda

1. Introduction
2. Preliminaries
3. Proposed Verification Framework
4. Evaluation Results
5. Conclusion

Agenda

DL-based Network Traffic Prediction

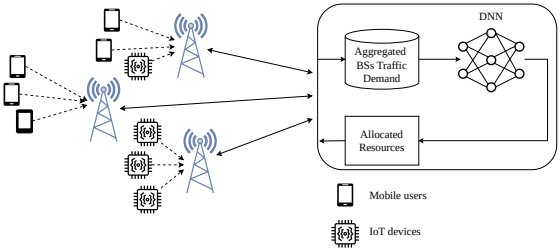


Figure: Deep learning (DL)-based network traffic prediction (NTP).

Pipeline description

- **Input:** historical traffic snapshots (*e.g.*, matrix/tensor of user demands at each base station)
- **DNN:** maps input $\rightarrow$ future demands
- **Application:** proactive link capacity provisioning, routing, traffic engineering

Can We Trust DL in Mission-Critical Tasks?

Autonomous Driving
Aircraft Autopiloting

Medical Equipments
AI-based Diagnosis

Security/Surveillance
Systems

[1]

[1] H. Zhang et al., *Formal verification of deep neural networks: Theory and practice*, Accessed: 2025-07-21, 2022. [Online]. Available: https://neural-network-verification.com/.

Solution – Related works

Adversarial Attacks on DL-based NTP

- Adversaries infiltrate smartphones or IoT devices
- Orchestrate botnets to inject minimal traffic
- Corrupting the DNN's capacity provisioning while evading anomaly detection

Existing Solutions – xAI

- xAI methods (e.g. GCAM, LRP)
- Identify input features and base stations susceptible to traffic-injection attacks;
- Inform adversarial training.

[a]

[a] S. M. Gholian et al., "Deexp: Revealing model vulnerabilities for spatio-temporal mobile traffic forecasting with explainable ai," *IEEE Transactions on Mobile Computing*, 2025.

Solution – NNV

Problem with non-certifying defense

- Non-certifying defense mechanisms are circumvented by stronger attack strategies
- Part of an ongoing arms race.

[a]

[a] A. Athalye et al., "Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples," in *International conference on machine learning*, PMLR, 2018, pp. 274–283.

Our Solution – NNV

- Neural networks verification (NNV) tools (e.g. NeuralSat, Reluplex, certified training)
- Provide provable guarantees that deep neural networks (DNN) remains resilient to attacks for all inputs within specified bounds.
- Complements explainable AI (xAI) and adversarial training
- Establishing DNN robustness through rigorous mathematical proofs

[a]

[a] H. Duong et al., "Neuralsat: A high-performance verification tool for deep neural networks," in *Computer Aided Verification*, Cham: Springer Nature Switzerland, 2025, pp. 409–423.

Agenda

System Model

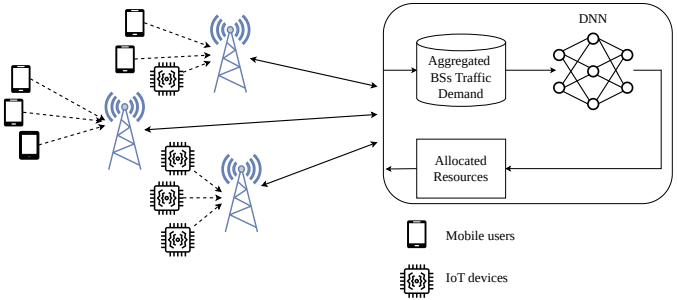


Figure: System model of DL-based NTP.

- A DNN is trained to forecast traffic at time t from T historical traffic snapshots $\delta^s(t-T), \ldots, \delta^s(t-1)$ (traffic demands at all base stations per slice s).
- **Input:** historical snapshots $\rightarrow$ 3D tensor (spatiotemporal).
- **Architecture:** encoder (3D-CNN) + decoder (MLP).
- **Output:** capacity forecast $c^s(t) = \{c_s^1(t), \ldots, c_s^M(t)\}$ at each base station
- M — number of base stations per slide s.

DNN architecture — DeepCog

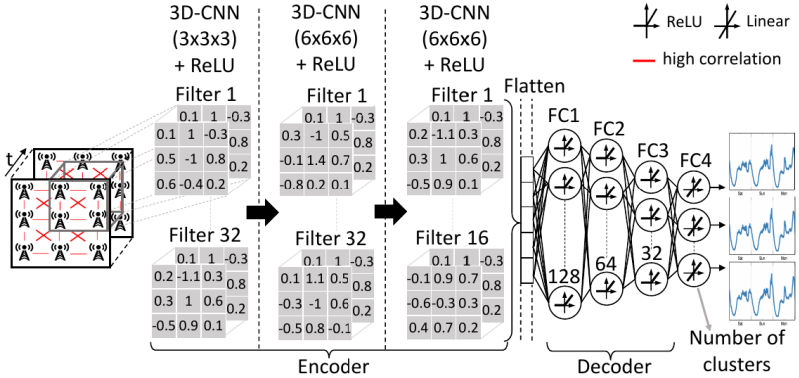


Figure: DeepCog[2] model based on 3D Conv.

[2] D. Bega et al., "Deepcog: Cognitive network management in sliced 5g networks with deep learning," in *IEEE INFOCOM 2019-IEEE conference on computer communications*, IEEE, 2019, pp. 280–288.

Threat Model

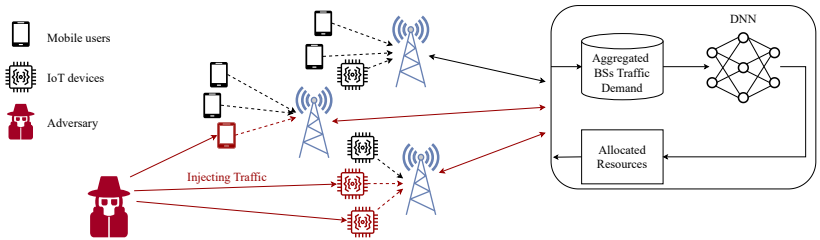


Figure: Threat model of DL-based NTP.

- Adversary controls compromised devices (*e.g.*, IoT, smartphones) that inject traffic into the network (Mirai-style).
- Goal: the DNN over-/under-provisioning → wasteful/service degradation.
- Injection is minimal to avoid detection and stay within data limits.

Agenda

Local Robustness Properties for DL-based NTP

Input property ϕ_{in}

Encodes *permitted adversarial traffic injection* as a hyperrectangle.

- Perturbation $\eta \in [\eta_L, \eta_U]$ (same dim as traffic snapshot $\delta^s(t)$).
- Adversarial input: $\delta'^s(t) = \delta^s(t) + \eta$.
- ϕ_{in} constrains perturbed traffic to $[\delta^s(t) + \eta_L, \delta^s(t) + \eta_U]$ so verification covers all injection scenarios within bounds.

Output property ϕ_{out}

Constrain forecast vs. actual demand $d_s^j(t)$ (tolerance ζ):

- **Overprovisioning:** $\neg(c_s^j(t) \geq (1+\zeta)\, d_s^j(t))$.
- **Underprovisioning:** $\neg(c_s^j(t) \leq (1-\zeta)\, d_s^j(t))$.

SoTA DNN Verifiers

NeuralSAT

Leverage SOTA DNN verifier used as a black-box; top-performing in Verification of Neural Networks Competition (VNNCOMP).

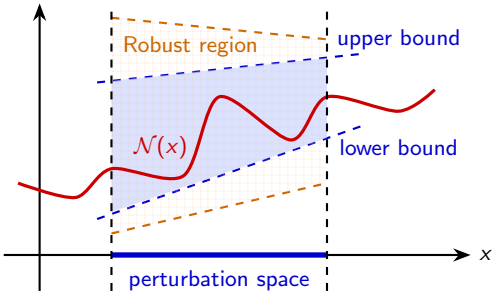


Figure: NNV propagates continuous intervals

Verification outcome

$\text{NEURALSAT}(\phi_{in}, \phi_{out}, \mathcal{N}(x)) \rightarrow$ outcomes:

- **SAT:** exists adversarial input in $[\eta_L, \eta_U]$ violating $\phi_{out} \Rightarrow$ vulnerable
- **UNSAT:** no such input $\Rightarrow$ proven robust
- **TIMEOUT:** DNN verifier unable to solve the instance within the time limit

H. Duong et al., "Neuralsat: A high-performance verification tool for deep neural networks," in *Computer Aided Verification*, Cham: Springer Nature Switzerland, 2025, pp. 409–423

Additional Conversion

Convert DL-based NTP to VNNCOMP-style inputs

1. 3DConv not supported by `NeuralSAT` → reimplement DeepCog as *two consecutive 2D* convolutional layers (no loss in forecast accuracy).
2. Export DeepCog to **ONNX**: open standard for interoperable DNN models; VNNCOMP uses ONNX for the network format[a].
3. Export ϕ_{in} (bounds $[\delta^s(t) + \eta_L, \delta^s(t) + \eta_U]$) and ϕ_{out} in **VNNLIB**: declarative format for input/output constraints in VNNCOMP.

[a] C. Brix et al., "The Fifth International Verification of Neural Networks Competition (VNN-COMP 2024): Summary and Results," *arXiv preprint arXiv:2412.19985*, 2024.

Agenda

Evaluation Setup

Dataset — Telecom Italia Milan 2014

- Publicly available mobile traffic data; 1,728 base stations.
- $\sim 7,000$ time steps, 50 days measurements, 10-minute granularity.
- *Internet activities* traffic demands [5].
- Spatial-temporal pattern well-suited for DeepCog-based traffic forecasting.

DNN Parameter — DeepCog

- 5×5 grid; $T = 10$ historical steps; each model forecasts capacity of the *central cell* only [2].
- $\alpha = 2$ (prioritize avoiding underprovisioning over overprovisioning) [5].
- Adam, learning rate 3×10^{-4}, 50 epochs; ReLU activation; 80:20 train-test split.
- Verify on perturbed data on test set, NeuralSAT timeout after 60s.

[a]

[a] Publicly available implementation at `https://github.com/thanhlexyz/nnv4ntp`

Robustness Analysis - Underestimation

	Percentage of injected traffic				
	+1%	+5%	+10%	+15%	+20%
$\zeta = 1\%$	0.0/1.0/0.0	0.0/1.0/0.0	0.0/1.0/0.0	0.0/1.0/0.0	0.0/1.0/0.0
$\zeta = 5\%$	0.0/1.0/0.0	0.0/1.0/0.0	0.0/1.0/0.0	0.0/1.0/0.0	0.0/1.0/0.0
$\zeta = 10\%$	0.0/1.0/0.0	0.0/1.0/0.0	0.0/1.0/0.0	0.0/1.0/0.0	0.0/1.0/0.0

Table: Fraction of UNSAT, SAT, TIMEOUT for underestimation of user demand

Observations

- Verifier consistently SAT across all injection levels (+1%–+20%) and all ζ (1%, 5%, 10%).
- Despite α set to reduce underestimation, the model remains vulnerable on all properties.
- Adversary can easily induce capacity underestimation beyond $\zeta \Rightarrow$ DoS / QoS degradation.

Robustness Analysis - Overestimation

	Percentage of injected traffic				
	+1%	+5%	+10%	+15%	+20%
$\zeta = 1\%$	1.0/0.0/0.0	1.0/0.0/0.0	1.0/0.0/0.0	0.0/1.0/0.0	0.0/1.0/0.0
$\zeta = 5\%$	1.0/0.0/0.0	1.0/0.0/0.0	1.0/0.0/0.0	0.0/0.0/1.0	0.0/1.0/0.0
$\zeta = 10\%$	1.0/0.0/0.0	1.0/0.0/0.0	1.0/0.0/0.0	1.0/0.0/0.0	0.0/0.0/1.0

Table: Fraction of UNSAT, SAT, TIMEOUT for overestimation of user demand

Observations

- **Small injection** (+1%, +5%, +10%): `NeuralSAT` often `UNSAT` ⇒ *provably robust* against overprovisioning.
- **High injection** (+15%, +20%): more `SAT`/`TIMEOUT` ⇒ extreme perturbations can compromise robustness.
- Takeaway: asymmetric profile — resilient to overestimation under moderate η; highly vulnerable to underestimation.

Agenda

Concluding remarks

- **Contribution:** Formal verification framework for DL-based mobile traffic prediction → provable robustness against adversarial traffic injection.
- **Approach:** Input perturbations (hyperrectangle) + output properties (over/underprovisioning bounds) + trained DNN → `NeuralSAT` for sound guarantees.
- **Finding:** Asymmetric robustness on Telecom Italia Milan: resilient to overestimation under moderate perturbations; highly vulnerable to underestimation attacks.
- **Future work:** Broader evaluation (more datasets and models); verifier in the adversarial retraining loop.