# Deep Learning with Requirements

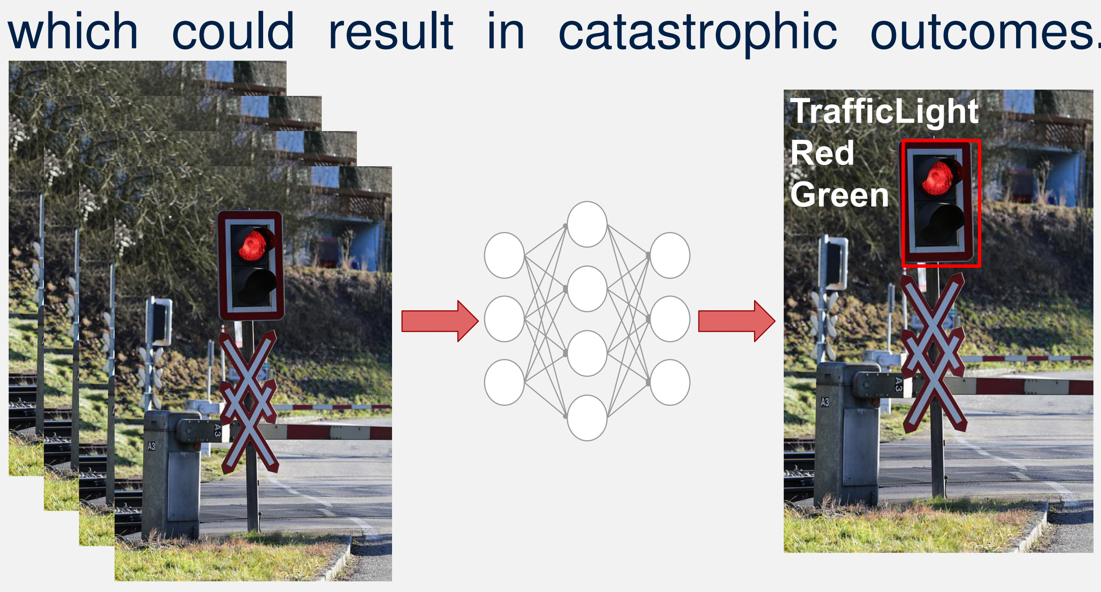
Eleonora Giunchiglia[1], Mihaela Cătălina Stoian[1], Thomas Lukasiewicz[2,1]
[1]University of Oxford, [2]TU Wien
{eleonora.giunchiglia,mihaela.stoian,thomas.lukasiewicz}@cs.ox.ac.uk

## Overview

Motivated by the need of having safe autonomous systems, especially in safety-critical scenarios, we introduce ROAD-R [1], the first real-world dataset equipped with logical **requirements** expressed as **logical constraints**.
We show that modern neural networks fail to satisfy the safety norms captured by the requirements and that we can develop methods to account for these constraints and simultaneously increase the models' performance.
The methods and ideas presented here are based on our paper "ROAD-R: The Autonomous Driving Dataset with Logical Requirements" [1] and on preliminary ongoing work.

## Example

Deep neural networks are increasingly becoming indispensable tools for autonomous systems, however, they might exhibit unexpected behaviour, which could result in catastrophic outcomes.

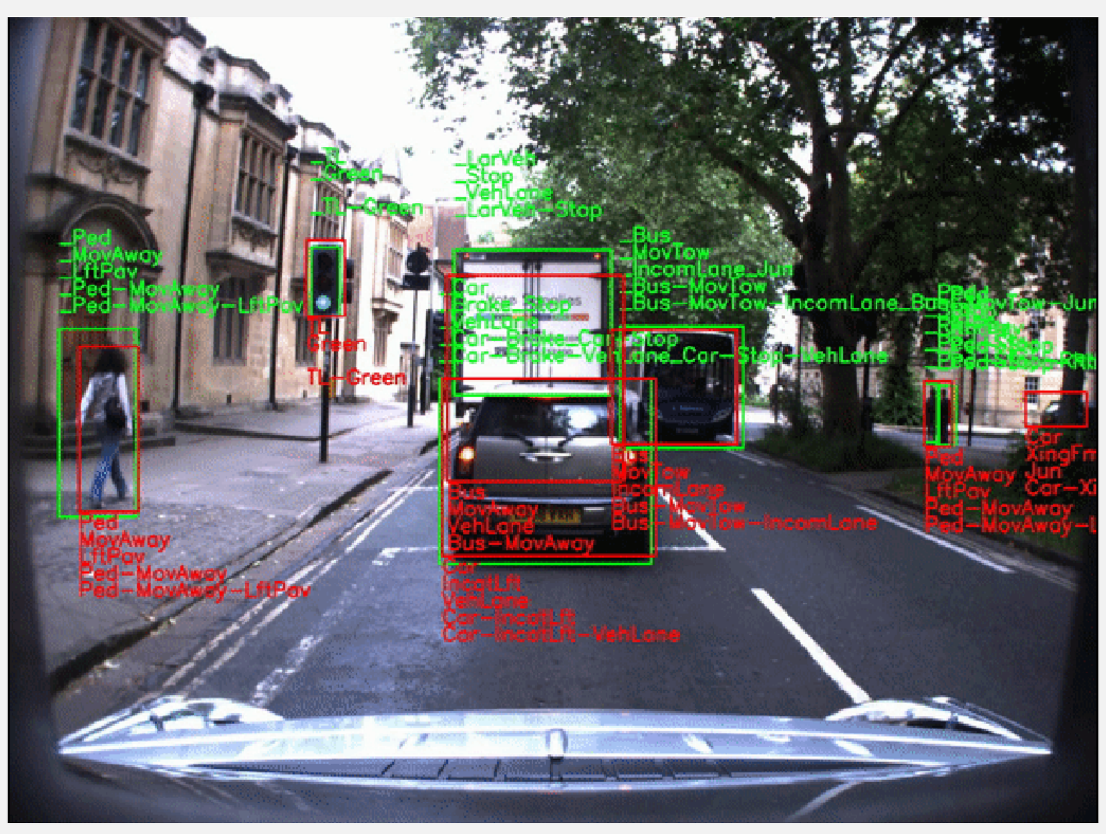



To avoid the above, **we advocate for the introduction of requirements elicitation as a standard step in deep learning models development**.

> Having requirements as additional input information could help making safer decisions!
> RedTrafficLight → ¬ GreenTrafficLight

## The ROAD-R dataset

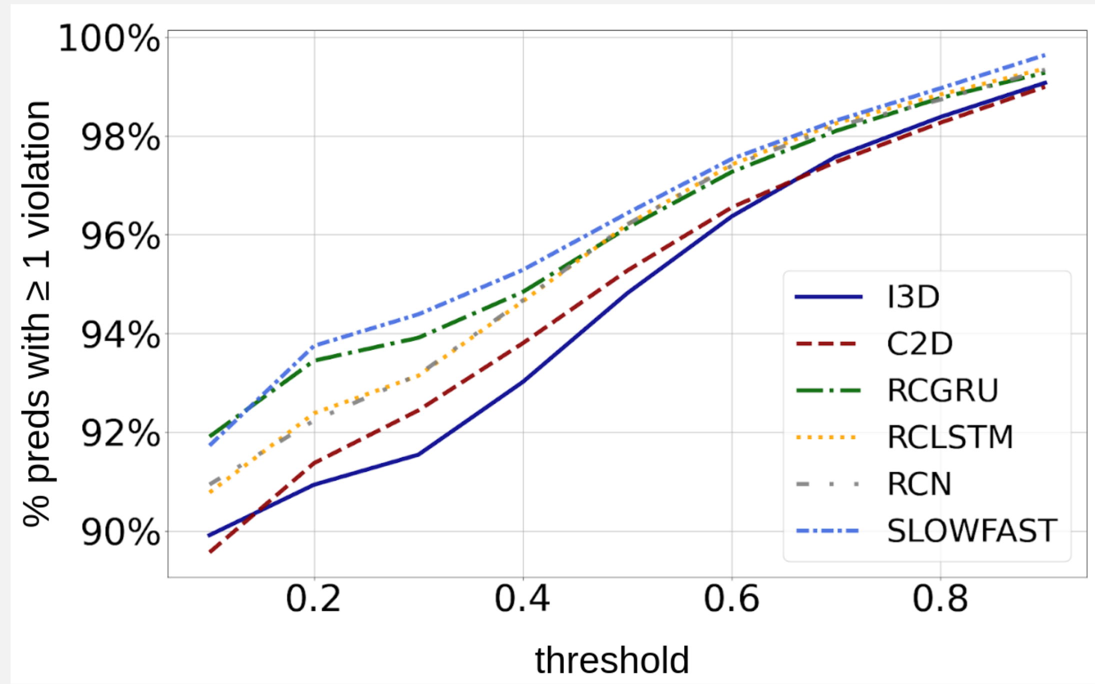
ROAD-R is the first publicly available dataset for autonomous driving manually annotated with requirements.



It extends the autonomous driving dataset ROAD [2] **with 243 logical constraints** expressing facts such as that a traffic light cannot be green and red at the same time.

$$\begin{aligned}\mathrm{RedTrafficLight} &\rightarrow \neg\mathrm{GreenTrafficLight}\\ \mathrm{MoveTowards} &\rightarrow \neg\mathrm{MoveAway}\\ \mathrm{VehicleLane} &\rightarrow \neg\mathrm{ParkingSpot}\\ \mathrm{Crossing} &\rightarrow \mathrm{Pedestrian} \vee \mathrm{Cyclist}\end{aligned}$$

## Modern NNs fail to satisfy safety norms




We picked 6 SOTA temporal feature learning architectures and incorporated them into the 3D-RetinaNet object detector [2]. After evaluating each model on ROAD-R, we noticed that:

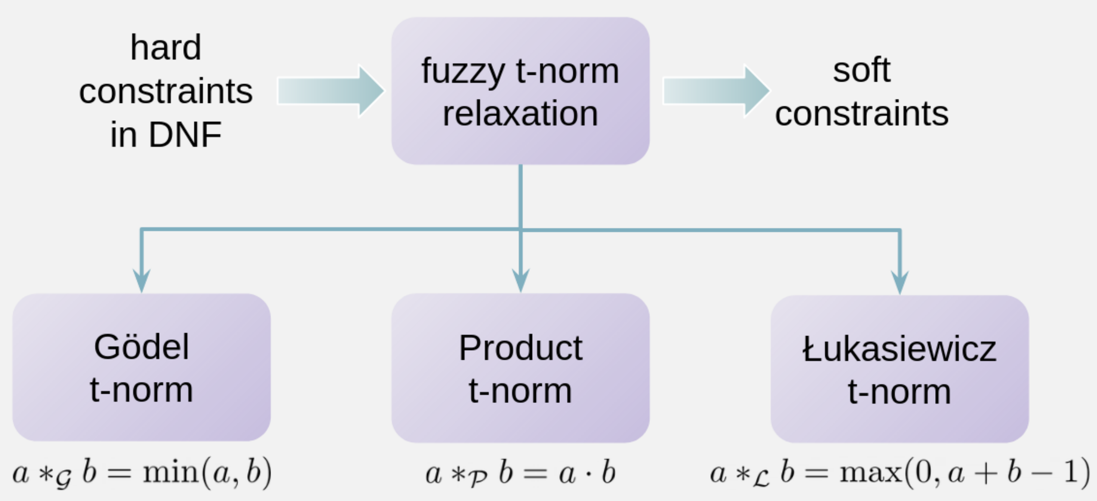
- most predictions violated at least one requirement,
- and about **99% of the predictions violated at least one constraint**, when using 0.9 for the threshold over which the prediction for a label becomes positive.

## Using logic to guide learning

We use logical constraints in two **complementary** ways to guide learning:

- to constrain the loss function of the neural networks during training,
- to correct the outputs of the neural networks during inference.

## Constraining the loss

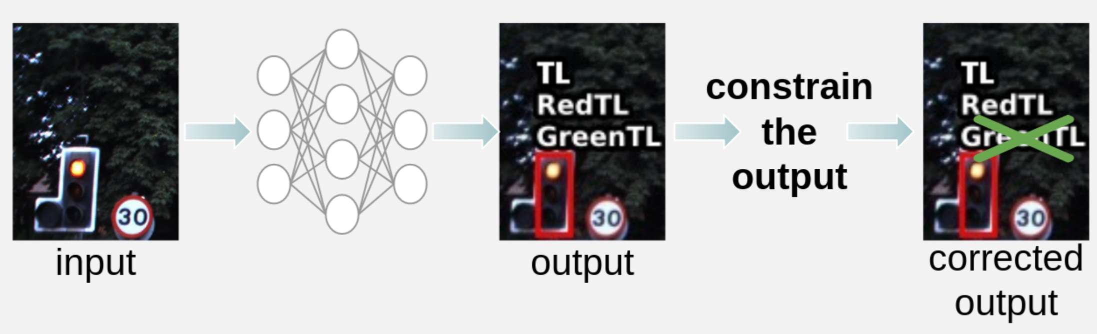



**Pros**: the domain knowledge is injected at training time and neural networks can learn from it.

**Cons**: no guarantee that constraints are satisfied.

> How to integrate the constraints in the loss?
>
> $$\mathrm{Loss} = \mathrm{Loss}_{loc} + \mathrm{Loss}_{cls} + \mathrm{Loss}_{requirements},$$
>
> where $\mathrm{Loss}_{loc}$ is the localisation loss of the candidate bounding boxes, $\mathrm{Loss}_{cls}$ is the classification loss, and $\mathrm{Loss}_{requirements}$ is a regularisation term indicating the degree of satisfaction of the relaxed constraints.

## Correcting neural outputs during inference

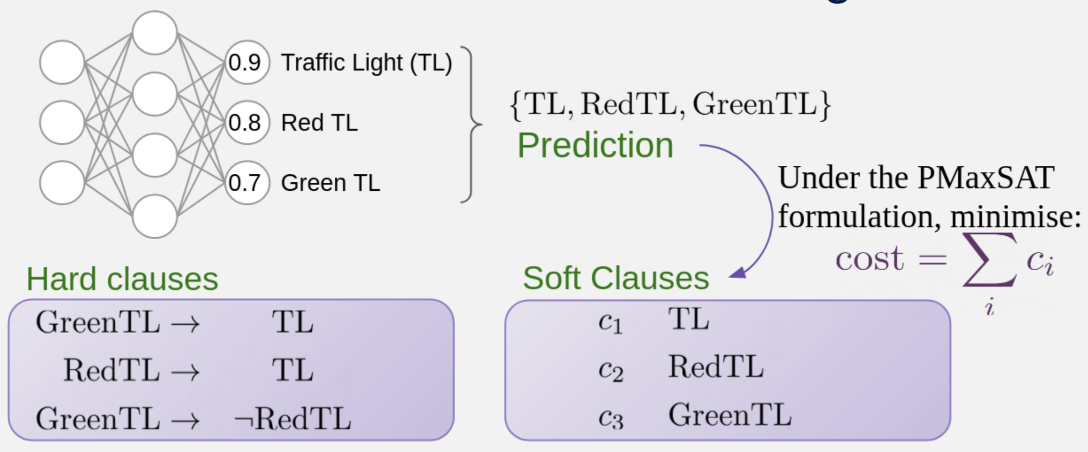



**Pros**: guaranteed constraint satisfaction.

**Cons**: the neural network cannot learn from the domain knowledge during training.

## Output correction as a PMaxSAT problem

Each label is assigned a positive cost for making a correction. The problem is then finding the optimal correction of a prediction and can be formulated as a PMaxSAT problem aiming to minimise the total cost of correcting all labels.




> It is better to correct the labels for which the model makes a lot of mistakes, so we assign the average precision of a label as its cost!
>
> $c_1 = AP_{TL}$
> $c_2 = AP_{RedTL}$
> $c_3 = AP_{GreenTL}$

## Results

Comparison of the frame mean average precision (f-mAP) between the standard models and the same models trained with the requirements loss and with post-processing.

| Model | Baseline | With Requirements |
|---|---|---|
| C2D | 27.57 | **28.16** (+0.59) |
| I3D | 30.12 | **31.21** (+1.09) |
| RCGRU | 30.78 | **31.81** (+1.03) |
| RCLSTM | 30.49 | **31.65** (+1.16) |
| RCN | 29.64 | **31.02** (+1.38) |
| SlowFast | 28.79 | **28.98** (+0.19) |

## Logic-guided semi-supervised learning

Additionally, we explored how logic can help when little annotated data is available and developed an approach for multilabel semi-supervised learning which:

- generates pseudo-labels for unlabelled data,
- and uses logic to improve their robustness.

Our method combines the principles of neurosymbolic integration with the standard self-training paradigm, where unlabelled samples are assigned pseudo-labels and are then treated as annotated data during training.
Below we show the preliminary results for different percentages of annotated ROAD-R data using 3D-RetinaNet with the RCGRU temporal feature learning architecture.

| Labelled Data (%) | Baseline | Logic-guided SSL |
|---|---|---|
| 10% | 15.12 | **21.98** (+6.86) |
| 20% | 20.22 | **24.98** (+4.76) |
| 50% | 27.08 | **27.88** (+0.80) |

## References

[1] Giunchiglia, E., Stoian, M. C., Khan, S., Cuzzolin, F., and Lukasiewicz, T. (2022). ROAD-R: The autonomous driving dataset with logical requirements. *Mach. Learn.* Accepted for publication.

[2] Singh, G., Akrigg, S., Maio, M. D., Fontana, V., Alitappeh, R. J., Saha, S., Saravi, K. J., Yousefi, F., Culley, J., Nicholson, T., Omokeowa, J., Khan, S., Grazioso, S., Bradley, A., Gironimo, G. D., and Cuzzolin, F. (2022). ROAD: The road event awareness dataset for autonomous driving. *TPAMI*.