Algorithmic Perspectives on Machine Learning Safety

Xiaowei Huang

Trustworthy Autonomous Cyber-physical Systems Lab,
University of Liverpool, UK

1.03 %

Outline

Motivations: **What does AI Safety constitute?**

Certification Framework – **F.E.V.E.R.**

- Falsification (through e.g., attacks, testing)
- Explanation
- Verification
- Enhancement (through e.g., training, regularisation, and randomisation)
- Reliability (through e.g., assessment, monitoring, and assurance)

Conclusions

Looking Ahead

- Distributed/Federated learning
- Foundation Models
- Energy Efficiency

Motivations: **What does AI Safety constitute?**

- trained on WPAFB 2009 dataset [11]: The images were taken by a camera system with six optical sensors that had already been stitched to cover a wide area of around 35km^2. Image size: 12,000$\times$10,000. The frame rate is 1.25Hz.

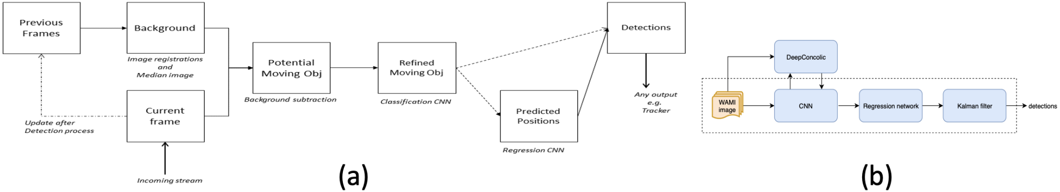


Figure: (a) The architecture of the vehicle detector. (b) Workflow for testing the WAMI tracking system.

[40] Reliability Validation of Learning Enabled Vehicle Tracking. ICRA2020

Safety of Learning Enabled Vehicle Tracking

Figure: Original detected tracks

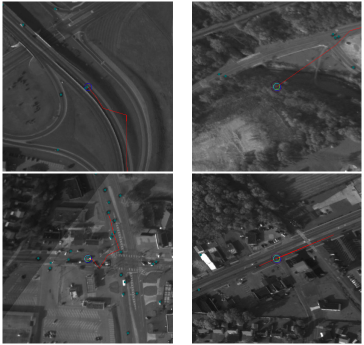


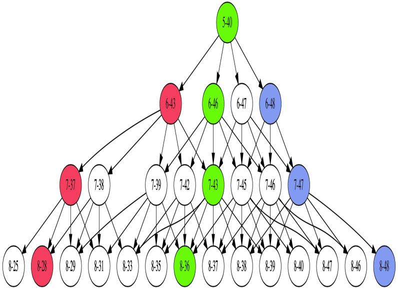
Figure: Distorted tracks

[40] *Reliability Validation of Learning Enabled Vehicle Tracking. ICRA2020*

Practical Verification of Vehicle Tracking System

(a) Heuristic search

(b) Verification



(c) Enumeration of all possible Tracks

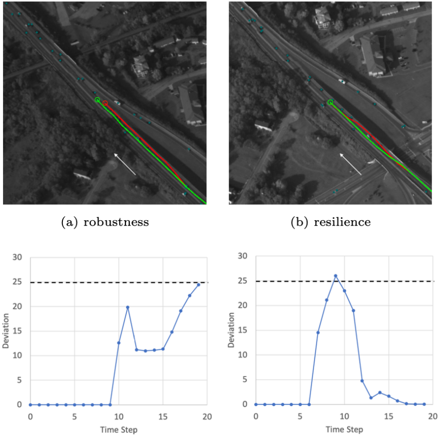
---

[24] Practical Verification of Neural Network Enabled State Estimation System for Robotics. IROS2020.

Verification of Robustness and Resilience

- robustness: consistently deliver its 'expected' functionality, even in the presence of disturbances to the input.
- resilience: withstand and recover from challenging conditions, which may involve internal failures and external shocks.

[23] Formal verification of robustness and resilience of learning-enabled state estimation systems for robotics. Neurocomputing, 2024.



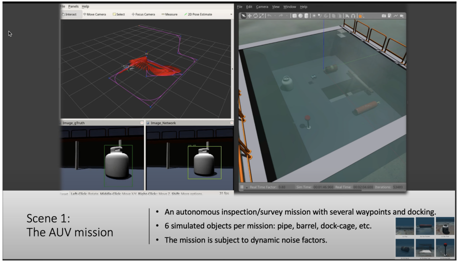
(a) robustness

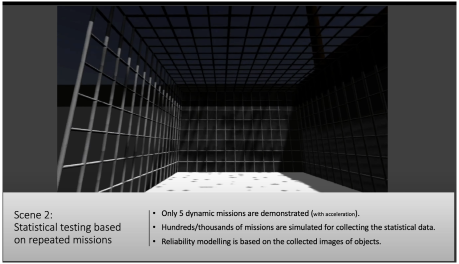
(b) resilience

UNIVERSITY OF LIVERPOOL

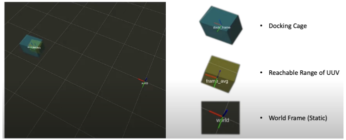
- Scenario: `https://youtu.be/akY8f5sSFpY?t=13`
- simulation / testing: `https://youtu.be/akY8f5sSFpY?t=155`
- verification: `https://youtu.be/WNjUP_qL6W4?t=475`

Underwater Vehicle

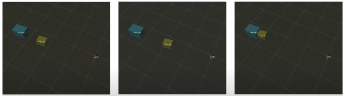




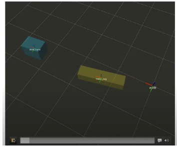
Case Demo without Crash

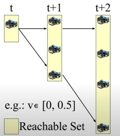


Case Demo without Crash



Case Demo with Crash





UNIVERSITY OF LIVERPOOL

Ground Vehicle

- https://www.youtube.com/watch?v=E95vh5sxs7I

AI Regulations, Whitepapers, Roadmaps, etc

- EU
  - GDPR [1], AI Act [8], Data Act [9]
- UK
  - Data Protection Act [2] and pro-innovative approach to regulate AI [10]
- US
  - Blueprint for an AI Bill of Rights [6] and AI Risk Management Framework [4]
- China
  - regulations for recommendation algorithms [5], deep synthesis [3], and algorithm registry [7]

EU AI Act

Different principles w.r.t. the risk levels:

1. unacceptable-risk AI: banned
2. high-risk AI:
   - human oversight,
   - technical robustness,
   - compliance with data protection rules,
   - appropriate explainability, non-discrimination and fairness,
   - social and environmental well-being
3. limited and minimal-risk:
   - transparency

Different principles w.r.t. the risk levels:

1. unacceptable-risk: banned
2. high-risk:
    - human oversight,
    - technical robustness,
    - compliance with data protection rules,
    - appropriate explainability, non-discrimination and fairness,
    - social and environmental well-being
3. limited and minimal-risk:
    - transparency

Translated into technical terms:

- robustness
- security
- privacy
- accountability
- fairness
- explainability
- safety
- human-centricity

Properties

Technical terms:

- robustness
- security
- privacy
- accountability
- fairness
- explainability
- safety
- human-centricity

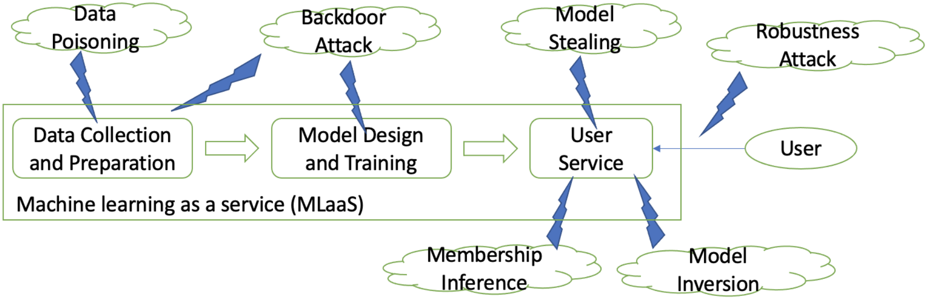
Known threats, e.g.,

- generalisation
- uncertainty
- robustness
- data poisoning
- backdoor
- model stealing
- membership inference
- model inversion

Formalised into **logical specifications** with statistical atomic propositions

[29] Bridging Formal Methods and Machine Learning with Global Optimisation. ICFEM, 2022.
[25] Machine Learning Safety. Springer, 2023.

Attack in ML Development Cycle



[25] Machine Learning Safety. Springer, 2023.

Trustworthiness = Certification (for **information**) + Explanation (for **communication**)

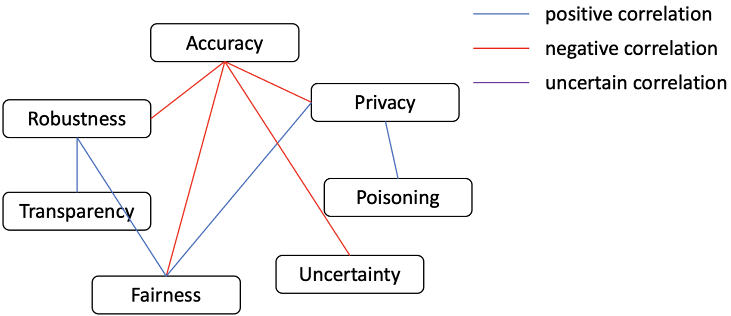
- Certification can be property-based, considering properties including safety, security, accountability, fairness, privacy, transparency, etc.
- Explanation is for the communication with stakeholders in a proper level of details.

Jump to outline

[26]: A Survey of Safety and Trustworthiness of Deep Neural Networks: Verification, Testing, Adversarial Attack and Defence, and Interpretability, Computer Science Review. 37 (2020): 100270.

Certification Framework – F.E.V.E.R.

Challenges: Complex Relations between Properties



- Incomplete, even for a given ML model
- Relations may change wrt dataset, model, etc

[15]: Building Guardrails for Large Language Models, ICML2024

Challenges: Environmental Uncertainty

- **Environmental noise (often white noise)**: may appear in all lifecycle stages: Data collection, Training, Inference
- **Distributional shift**: AI model may work on many environments/domains that are different from the enviroment where the training data was collected
- **Adversarial/malicious attacker**: Different attacks (robustness, backdoor, privacy, etc) may appear on different lifecycle stages
- **Human misbehaviour**: "A whopping 99 percent of autonomous vehicles accidents were caused by human error", a new report from IDTechEx shows.

Challenges: Others

- Model Complexity: size, complexity, dynamic update, imperfect information
- Properties: not well defined, or undefined
- Certification techniques: lack of novel techniques

For example:

- **Robustness:** $\phi_{rob}(\mathbf{w}, \mathbf{x}) \triangleq \Box(\textbf{inference} \Rightarrow \phi^1_{rob}(\mathbf{w}, \mathbf{x}))$

where $\phi^1_{rob}(\mathbf{w}, \mathbf{x}) \triangleq \forall \mathbf{r} : ||\mathbf{r}||_2 \leq c \Rightarrow |P(Y|\mathbf{x}+\mathbf{r}, \mathbf{w})(\hat{y}) - P(Y|\mathbf{x}, \mathbf{w})(\hat{y})| \leq \epsilon_{rob}$

- **Backdoor:**
  $\phi_{bac}(\mathbf{w}, \mathbf{d}_{train}, \mathbf{d}_{adv}) \triangleq \neg\Diamond(training \wedge \phi^2_{bac}(\mathbf{d}_{train}) \wedge \neg\phi^2_{bac}(\mathbf{d}_{train} \cup \mathbf{d}_{adv}))$

where $\phi^1_{bac}(\mathbf{w}) \triangleq \neg\exists\mathbf{r}\forall\mathbf{x}\forall y : P(Y|\mathbf{x}+\mathbf{r}, \mathbf{w})(y_{adv}) \geq P(Y|\mathbf{x}+\mathbf{r}, \mathbf{w})(y)$ and $\phi^2_{bac}(\mathbf{d}) \triangleq \neg\exists\mathbf{r}\forall\mathbf{x}\forall y : \mathbb{E}_{\mathbf{w}\sim P(W|\mathbf{d})}(P(Y|\mathbf{x}+\mathbf{r}, \mathbf{w})(y_{adv})) \geq \mathbb{E}_{\mathbf{w}\sim P(W|\mathbf{d})}(P(Y|\mathbf{x}+\mathbf{r}, \mathbf{w})(y))$. It expresses that, there does not exist any time in the future that the model is resistant to the backdoor trigger if trained on the usual training dataset but is not resistant if trained on the poisoned dataset.

---

[29] Bridging Formal Methods and Machine Learning with Global Optimisation. ICFEM 2022 (keynote and invited paper) & Journal of Logical and Algebraic Methods in Programming, 2023.

Specification Language

We end up have to deal with several probabilistic atoms such as

- Posterior Distribution $P(W|\mathbf{d})$
- Data Distribution $\mathcal{D}$
- Distribution of Predictive Labels $P(\hat{Y}|\mathbf{d}, \mathbf{w})$
- distance between distributions such as $D_{KL}(\mu, \mu)$ or $||\mu - \mu||_p$

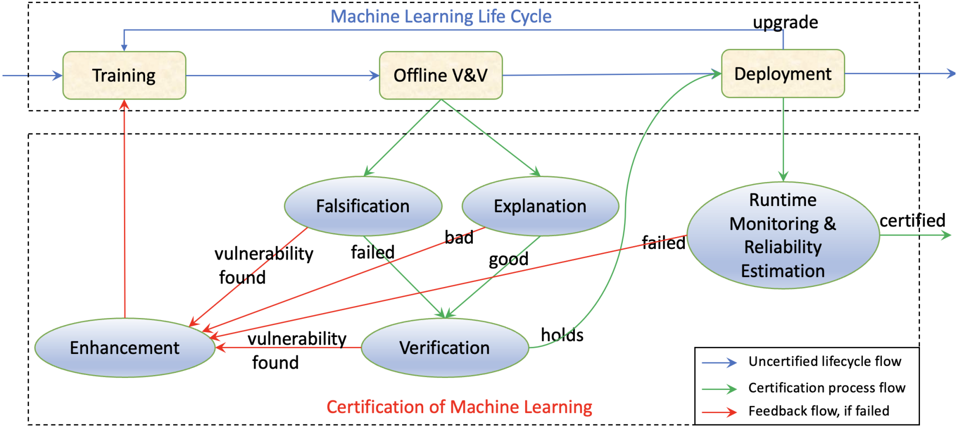
Nevertheless, the most tricky part (and the most drastic difference with existing safety critical software) is

- Environmental uncertainty, and
- Dynamic evolution of learning

It can be impossible to write a complete specification by human experts. How to deal with this?

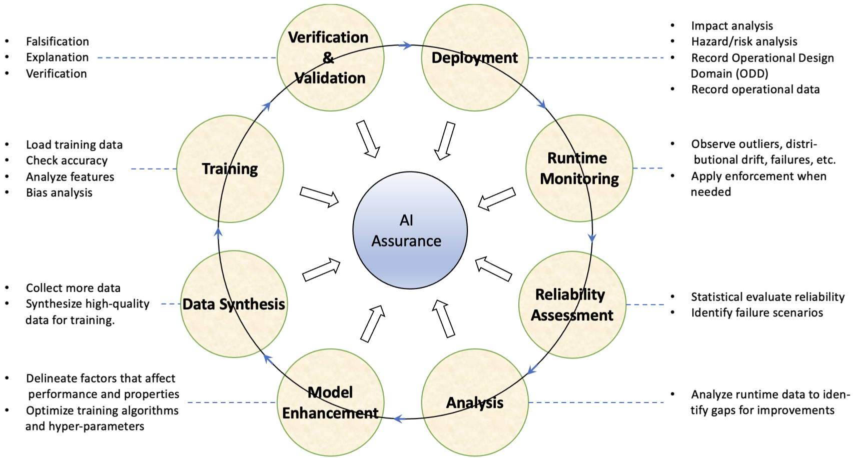
---

[29] Bridging Formal Methods and Machine Learning with Global Optimisation. ICFEM 2022 (keynote and invited paper) & Journal of Logical and Algebraic Methods in Programming, 2023.



[26] A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability. Computer Science Survey, 2020

Lifecycle



[25] Machine Learning Safety. Springer, 2023.

Assurance is a description of what high-quality software *development processes* should be put in-place to create (safety-critical) software that performs its desired function.

If *life cycle evidence* can be produced to demonstrate that these processes have been correctly and appropriately implemented, then such software should be assured.

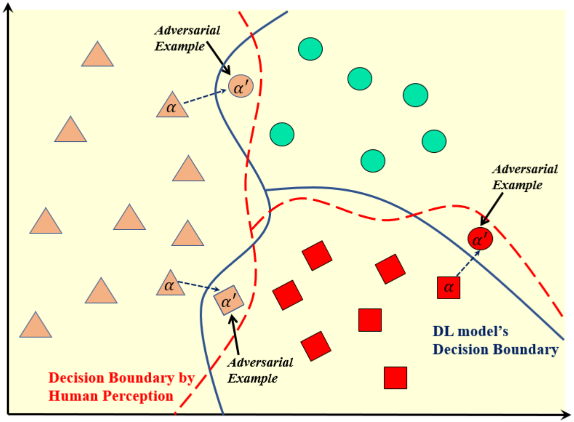
leads to software standards such as

- DO-178B/C, Software Considerations in Airborne Systems and Equipment Certification
- ISO 26262: standards for the functional safety of road vehicles

Falsification aims to find evidence to demonstrate the weaknesses of a trained machine learning model or a machine learning training process. Popular techniques include

- adversarial attack
- testing
- Monte Carlo sampling based methods,
- genetic algorithm based methods,
- etc

Robustness Error



DL model: classifies α and α' **differently**

Human: should remain the **same**

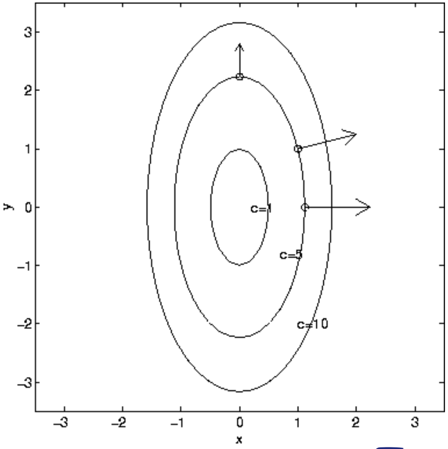
For robustness, one of earliest adversarial attack : optimization based formulation with L_2-norm metric

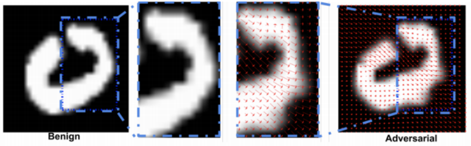
- Model $f : \mathbb{R}^{s_1} \rightarrow \{1 \ldots s_K\}$ with s_K labels
- $x \in \mathbb{R}^{s_1} = [0,1]^{s_1}$ is an input
- $t \in \{1 \ldots s_K\}$ is a target misclassification label

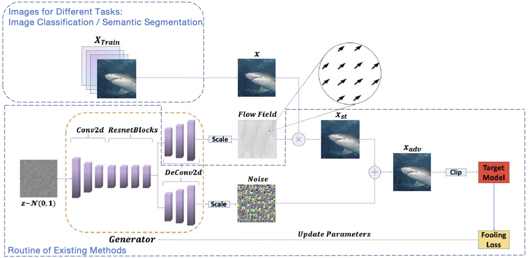
Find the adversarial perturbation r via

$$\begin{aligned} & \min ||r||_2 && \text{assure human-decision unchanged} \\ \textit{s.t.} \quad & \arg\max_l f_l(x+r) = t && \text{assure misclassification} \\ & x + r \in \mathbb{R}^{s_1} && \text{assure perturbed image feasible} \end{aligned} \tag{1}$$

The gradient vector $\nabla f(x, y)$ points in the direction of greatest rate of increase of $f(x, y)$





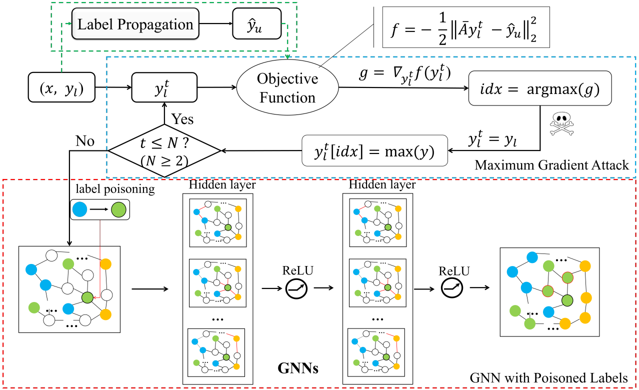


- Instead of perturbing the pixel values, adversarial attacks can be achieved by **spatial transformation** – on MNIST: digit "0" is misclassified as "2" (left figure)
- Different metric is required to measure pixel's **spatial displacement**
- Perturb spatial location and values of pixels simultaneously on **a set of images**?

[39] Generalizing Universal Adversarial Perturbations for DNNs. ICDM2020 & Machine Learning, 2023

Label Poisoning Attack on Graph Neural Networks

1. label propagation to generate predictive labels
2. maximum gradient attack to poison data labels
3. GNN training with poisoned labels



[33] *Adversarial Label Poisoning Attack on Graph Neural Networks via Label Propagation. ECCV2022*

Attacking Large Language Models

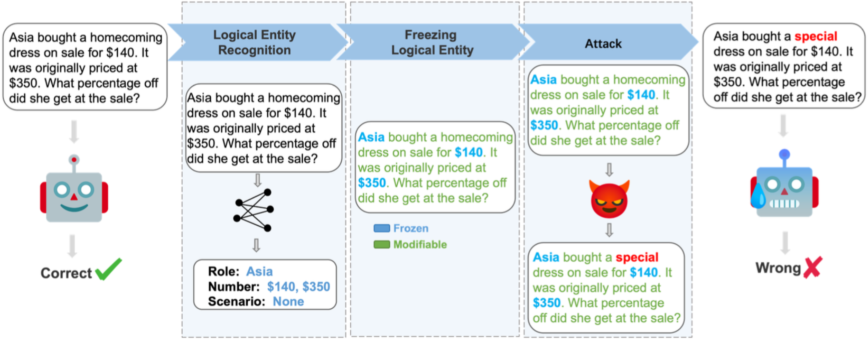


Figure 2: The overview of MathAttack. First, we utilize an NER model to identify logical entities. Then we freeze the logical entities, preventing the attacker from modifying them. Finally, we utilize word-level attacker to attack the LLMs while not changing those frozen logical entities.

[51] MathAttack: Attacking Large Language Models towards Math Solving Ability. AAAI2024

UNIVERSITY OF LIVERPOOL

- Well established in many industrial standard for software used in safety critical systems, such as ISO26262 for automotive systems and DO 178B/C for avionic systems.

- Coverage-guided testing
  - (step 1) generate as many as possible the test cases according to the structural information of the model, and
  - (step 2) use the test cases to evaluate if the model performs well with respect to certain properties

Coverage-Guided Testing

- Coverage Metrics
  - Structural Coverage, e.g., MC/DC coverage metrics [38] (Core idea: not only the presence of a feature needs to be tested but also the causal effects of less complex features on a more complex feature must be tested.)
  - Scenario Coverage
- Test Case Generation Methods
  - Fuzzing
  - Symbolic/Concolic execution [39], etc

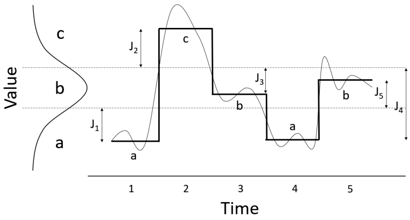
- check DeepConcolic: `https://github.com/TrustAI/DeepConcolic`

[38] Structural Test Coverage Criteria for Deep Neural Networks. ICSE2019

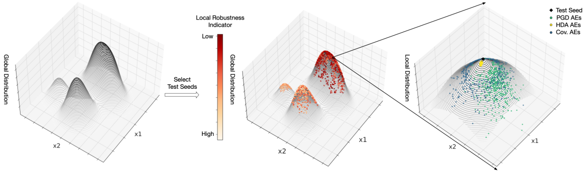
[39] Concolic Testing for Deep Neural Networks. ASE2018

More Advanced Testing

Coverage-Guided Testing for Recurrent Neural Networks [20]



Hierarchical Distribution-Aware Testing of Deep Learning [21]



Jump to outline

[20] Coverage-Guided Testing for Recurrent Neural Networks. IEEE trans. on Reliability, 2021

[21] Hierarchical Distribution-Aware Testing of Deep Learning. ACM Trans. on Software Engineering and Methodology, 2023

Explanation

The black-box nature of deep neural networks (DNNs) makes it impossible to understand why a particular output is produced, creating demand for "Explainable AI".

(a) 'cowboy hat' (b) 'dog' (c) 'numbfish' (d) 'sheep'

Figure: Input images and explanations from PROTOZOAfor Xception (red labels highlight misclassification or counter-intuitive explanations) [37]

For certification, we need not only correct classification but also correct explanation.

[37] Explaining Image Classifiers using Statistical Fault Localization. ECCV2020

Explanation through Statistical Fault Localization

Adopting the definition of explanations by Halpern and Pearl, which is based on their definition of actual causality. What we required:

1. an explanation is a *sufficient* cause of the outcome;
2. an explanation is a *minimal* such cause (that is, it does not contain irrelevant or redundant elements);
3. an explanation is *not obvious*; in other words, before being given the explanation, the user could conceivably imagine other explanations for the outcome.

What we propose:

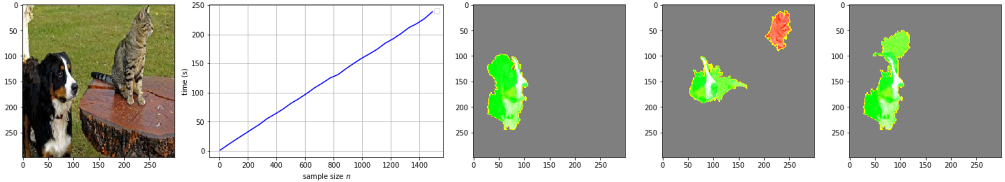
- SFL (stochastic fault localisation) measures to rank the set of pixels of x by slightly abusing the notions of passing and failing tests

[37] Explaining Image Classifiers using Statistical Fault Localization. ECCV2020

BayLIME: Bayesian LIME

Utilising Bayesian variant to deal with

- consistency in repeated explanations of a single prediction (as shown below, with LIME, different explanations can be generated for the same prediction)



(A) (B) (C) (D) (E)

- explanation fidelity
- robustness to kernel settings

[50] BayLIME: Bayesian Local Interpretable Model-Agnostic Explanations. UAI2021

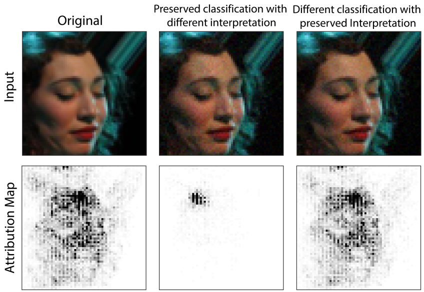


Figure: Two types of misinterpretations after perturbation

Novel black-box evaluation methods:

- based on Genetic Algorithm
- for both *worst-case* and *overall* robustness of explanations
- new interpretation Discrepancy Metrics

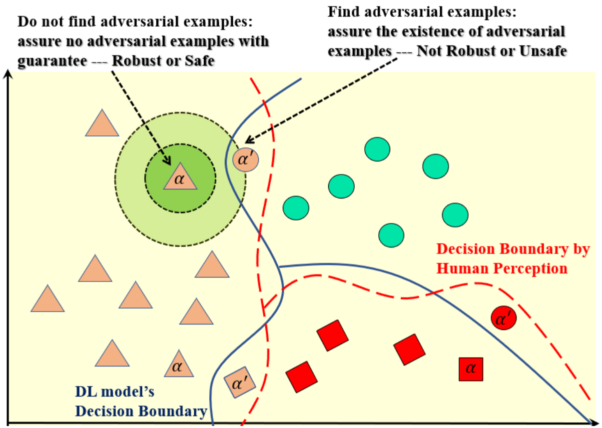
Jump to outline

[22] SAFARI: Versatile and Efficient Evaluations for Robustness of Interpretability. ICCV2023

Verification

Verification aims to determine if a model satisfies certain properties. Popular techniques include

- reduction to constraint solving
- over-approximation
- global optimisation based methods
- statistical evaluation
- randomised smoothing
- etc



(Robustness) Verification: verify if a certain input area can exclude misclasssification with **guarantees**

Verification by Reduction to Constraint Solving

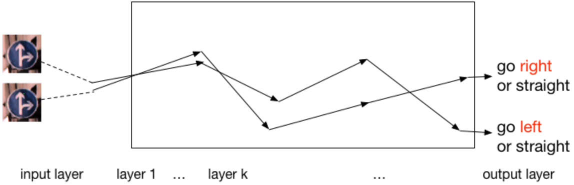
- (step 1) encode the entire network
- (step 2) encode the robustness constraint over the input
- (step 3) compute the result by solving the constraints

- encode the network
- Let $\vec{t}_{i+1}$ have value 0 or 1 in its entries and have the same dimension as $\vec{v}_{i+1}$, and M be a very large constant number that can be treated as ∞.
- we have the following MILP constraints for every layer $i = 1..K-2$

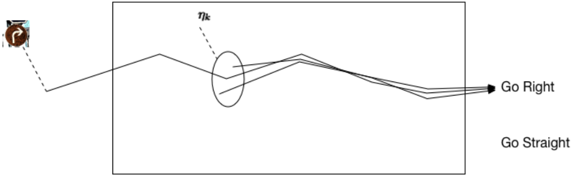
$$
\begin{aligned}
\vec{v}_{i+1} &\geq \mathbf{W}_i \vec{v}_i + \vec{b}_i, \\
\vec{v}_{i+1} &\leq \mathbf{W}_i \vec{v}_i + \vec{b}_i + M\vec{t}_{i+1}, \\
\vec{v}_{i+1} &\geq \mathbf{0}, \\
\vec{v}_{i+1} &\leq M(1 - \vec{t}_{i+1}),
\end{aligned}
\tag{2}
$$

Verification by Region Propagation

How does neural network process (two very similar) inputs?



How does verification work?



A layer-by-layer explicit search with SMT solver

---

[27] Safety verification of deep neural networks. CAV2017

Verification by Global Optimisation

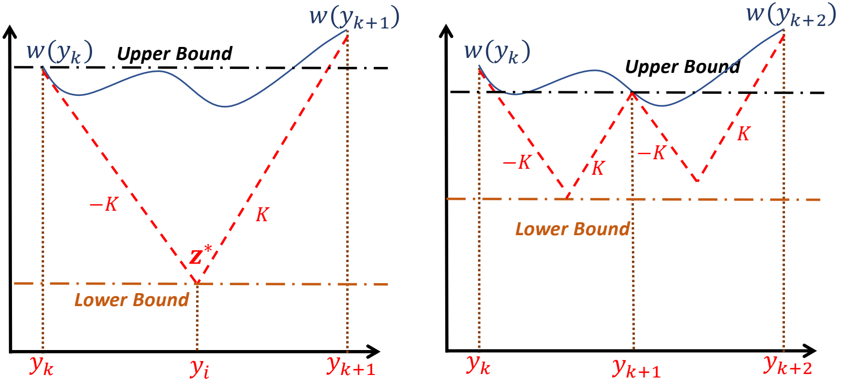


Figure: A lower-bound function designed via Lipschitz constant

[35] Reachability Analysis of Deep Neural Networks with Provable Guarantees. IJCAI2018

- Reduction to Monte-Carlo Tree Based Search
- Reduction to Other Global Optimisation Method
- Reduction to Two-player Game

[42] Feature-guided black-box safety testing of deep neural networks. TACAS2018.
[36] Global robustness evaluation of deep neural networks with provable guarantees for the Hamming distance. IJCAI2019

[43] A game-based approximate verification of deep neural networks with provable guarantees.

Theoretical Computer Science, 2020.

- Scalability
- Mostly work with Robustness
- Can only deal with deterministic variables/neurons, but machine learning problems are mostly statistical ...

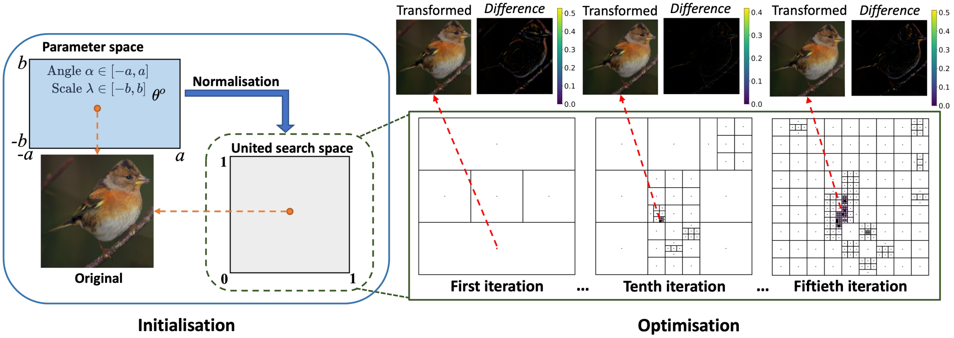


Figure: After normalising the parameter space to a unit search space, GeoRobust performs a sequence of space divisions to find the global worst-case transformation.

[41] Towards Verifying the Geometric Robustness of Large-scale Neural Networks. IJCAI2023

- Based on randomised smoothing
- black-box certification
- a novel approach based on the generalisation theorem between distributions
- by employing f-divergence to quantify the distance between distributions, our approach can be expanded to provide certification for a range of l_p-norm bounded perturbations

[34] Reward Certification for Policy Smoothed Reinforcement Learning. AAAI2024

New Challenges

- needs to compare a pair of inputs, rather than a single one
- Queries are too slow

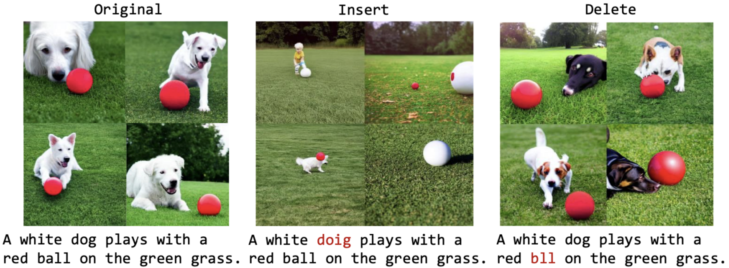
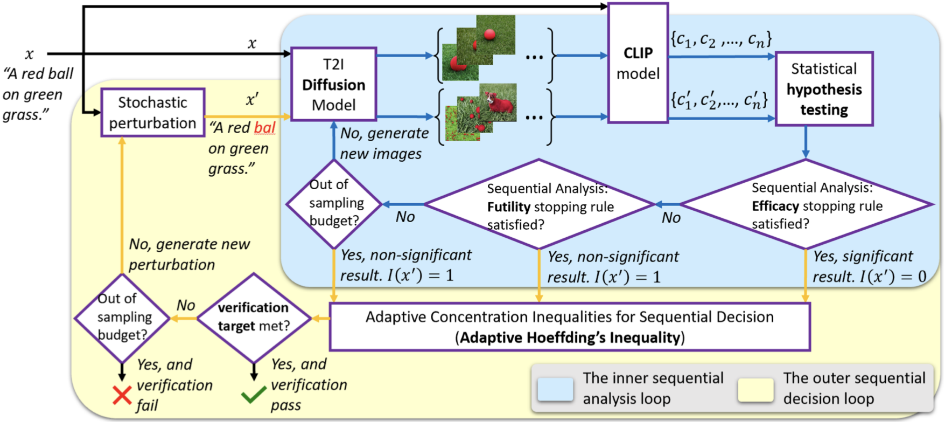


Fig. 1: Examples illustrating perturbations applied to the prompt for Stable Diffusion, employing two methods as described in Sec. 3.2

[45] ProTIP: Probabilistic Robustness Verification on Text-to-Image Diffusion Models against Stochastic Perturbation. ArXiv, 2024

Statistical Verification on Text-to-Image Diffusion Models



Jump to outline

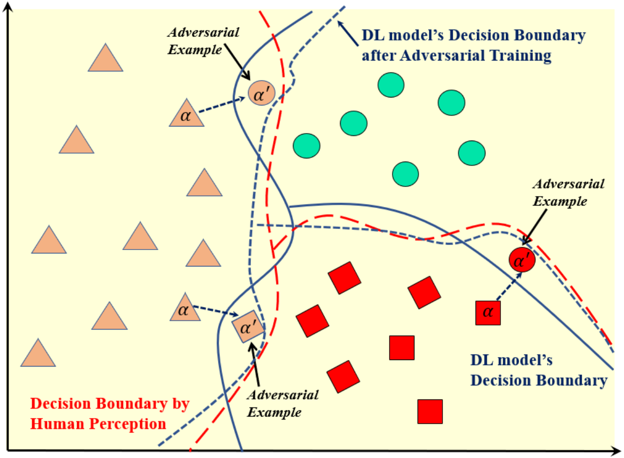
[45] ProTIP: Probabilistic Robustness Verification on Text-to-Image Diffusion Models against Stochastic Perturbation. ArXiv, 2024

Enhancement through Rectification

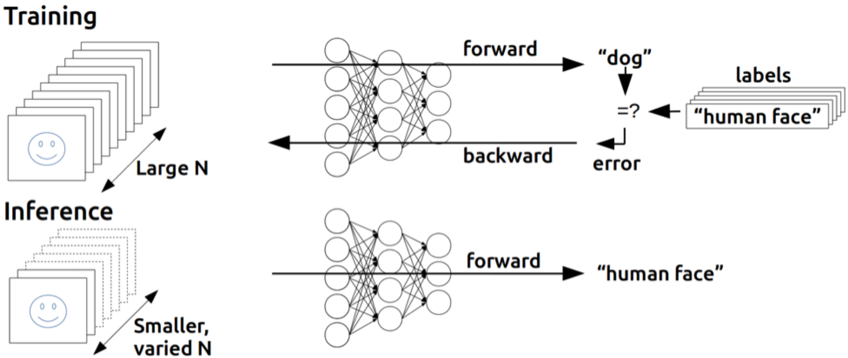
Rectification aims to enhance the machine learning training process or the trained machine learning model, so that the resulting machine learning model performs better with respect to the properties. Popular techniques include

- adversarial training
- regularisation
- outlier detection
- randomisation (based on differential privacy)
- etc

Model Improvement for Robustness

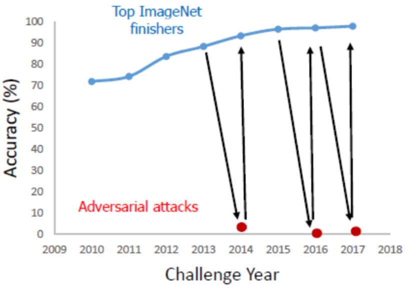


Training and Inference of Deep Learning



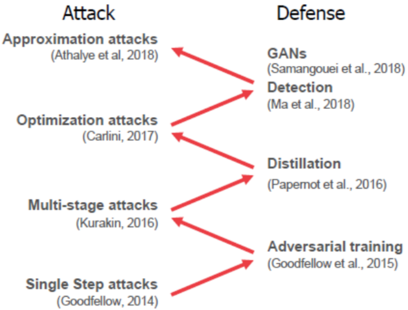
Attack vs. Defence: An Endless Game

Adversarial attacks cause a catastrophic reduction in ML capability



ImageNet Classification

Many defenses have been tried and failed to generalize to new attacks



Attack / Defense Cycle

@ DARPA's GARD programme

Consider weight correlation during the training

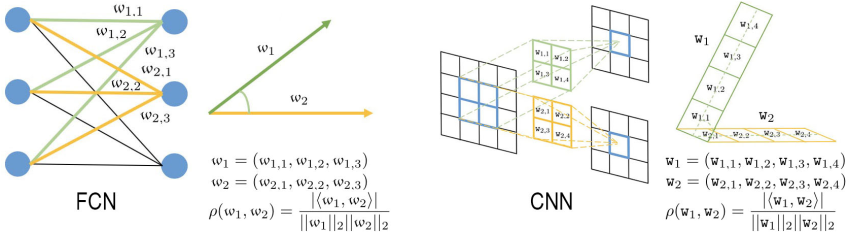


Figure: For fully connected networks, the weight correlation of any two neurons is the cosine similarity of the associated weight vectors. For convolutional neural networks, the weight correlation of any two filters is the cosine similarity of the reshaped filter matrices.

[32] How does Weight Correlation Affect Generalisation Ability of DNNs? NeurIPS2020

PAC-Bayes Bound

(McAllester, 1999) considers a generalization bound on the parameters

Posteriori distribution Q on parameters Θ

Priori distribution P on parameters Θ

$$\mathbb{E}_{\Theta\sim Q}[\mathcal{L}_D(f_\Theta)] \leq \mathbb{E}_{\Theta\sim Q}[\mathcal{L}_S(f_\Theta)] + \sqrt{\frac{\mathrm{KL}(Q||P) + \log\frac{m}{\delta}}{2(m-1)}}$$

Likelihood δ

Expected loss on input space D

Expected loss on samples S from D

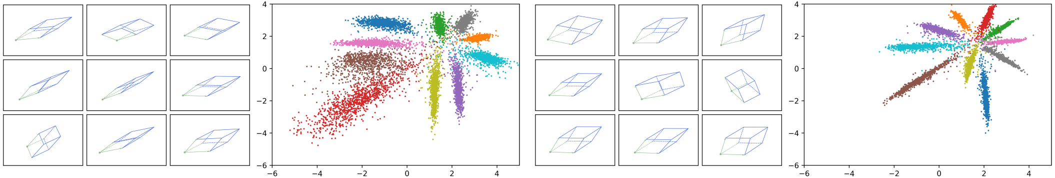
Number of samples

KL divergence plays a key role in the generalization bound

- a small KL term will help tighten the bound
- a larger KL term will loose the bound

[31] How does Weight Correlation Affect Generalisation Ability of DNNs? NeurIPS2020

Weight Expansion Helps Generalisation



(a) Normal CNN weight volume: 0.1592 (b) Normal CNN feature visualization (c) Dropout CNN weight volume: 0.3239 (d) Dropout CNN feature visualization

Figure: Visualization of weight volume and features of the last layer in a CNN on MNIST, with and without dropout during training

---

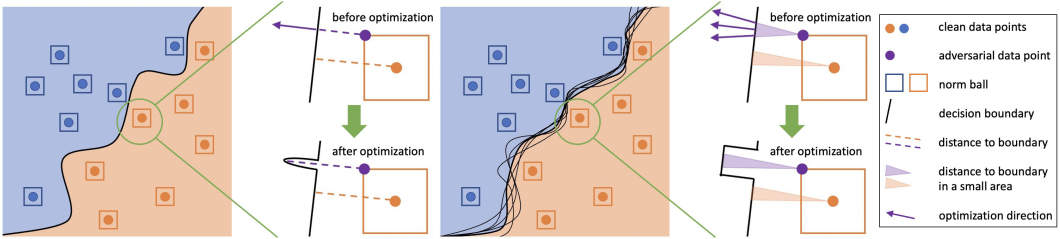
[32] Weight Expansion: A New Perspective on Dropout and Generalization. Transactions on Machine Learning Research. 2022

- treating model weights as random variables allows for enhancing adversarial training through **S**econd-Order **S**tatistics **O**ptimization (S^2O) with respect to the weights
- derive an improved PAC-Bayesian adversarial generalization bound, which suggests that optimizing second-order statistics of weights can effectively tighten the bound.
- through experiments, we show that S^2O not only improves the robustness and generalization of the trained neural networks when used in isolation, but also integrates easily in state-of-the-art adversarial training techniques like TRADES, AWP, MART, and AVMixup, leading to a measurable improvement of these techniques.

[30] Enhancing Adversarial Training with Second-Order Statistics of Weights. CVPR2022

Probabilistic View Helps Training

- embedding neural network weights with random noise
- utilize Taylor series to expand the objective function over weights (e.g., zeroth term, first term, second term, etc).



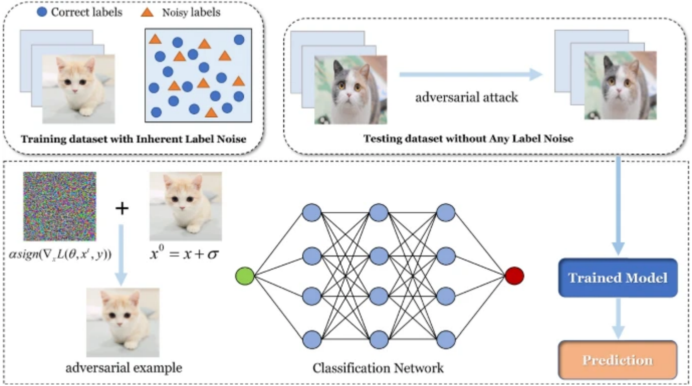
(a) Adversarial Training of TRADES (b) Our Method

[30] Randomized Adversarial Training via Taylor Expansion. CVPR2023

Most AT methods do not take into account the presence of noisy labels.

We consider two essential metrics in AT:

- trade-off between natural and robust accuracy;
- robust overfitting



[30] *Nrat: towards adversarial training with inherent label noise. Machine Learning, 2024*

- Robust Representation Training: learns representations that capture only task-relevant information based on the bisimulation metric of states.
- Semi-Contrastive Representation attack
- Adversarial Representation Tactics, which combines Semi-Contrastive Adversarial Augmentation with Sensitivity-Aware Regularizer to improve the adversarial robustness

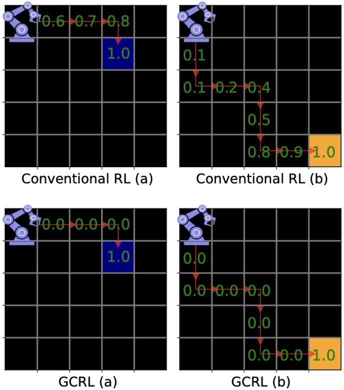
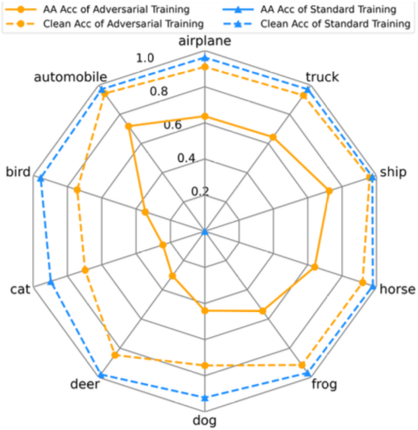


Figure 1: Trajectories of the agent at state s approaching blue and orange goals in conventional RL and GCRL, where the designated goals vary with different initialization. Rewards are indicated in each block along the trajectories.

[44] *Representation-Based Robustness in Goal-Conditioned Reinforcement Learning. AAAI-2024*

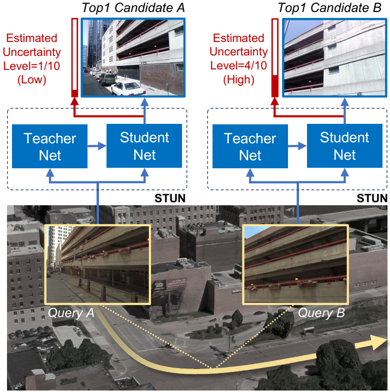
- Instead of average robustness, assessing worst-case robustness, avoiding robustness against categories like inanimate objects (with high accuracy) while vulnerable to crucial categories such as “human” (with low accuracy).
- adversarial training as a min-max- max framework, to ensure both robustness and fairness of the trained model



[46] Towards Fairness-Aware Adversarial Learning. CVPR2024

1. train a teacher net
2. supervised by the pretrained teacher net, a student net with an additional variance branch is trained
3. During the online inference phase, we only use the student net to generate both a place prediction and the uncertainty

This can not only generate uncertainty for each prediction but also improve the accuracy (i.e., generalisation).

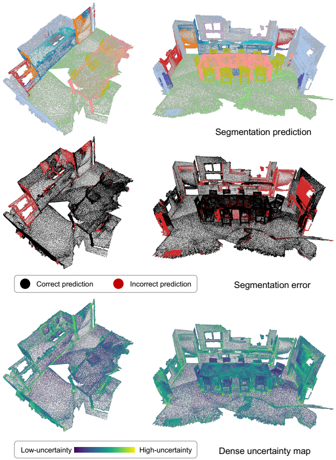


[12] *STUN: Self-Teaching Uncertainty Estimation for Place Recognition. IROS2022*

- building a probabilistic embedding model and then
- enforcing metric alignments of massive points in the embedding space

Figure 1 for 3D semantic segmentation. We have segmentation prediction (top), segmentation error (middle) and dense uncertainty map (bottom) of two scenes from ScanNet.

- Incorrect predictions tend to have high uncertainties.



[13] Uncertainty Estimation for 3D Dense Prediction via Cross-Point Embeddings. RA-L. 2023

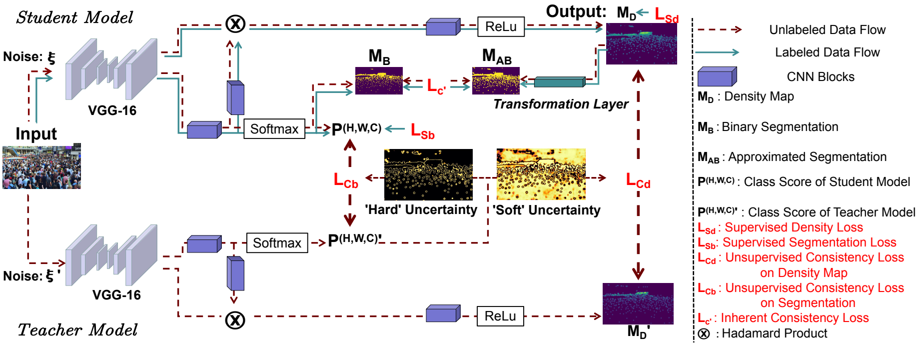


Figure: The pipeline of our uncertainty-aware framework for semi-supervised crowd counting.

[13] Spatial Uncertainty-Aware Semi-Supervised Crowd Counting. ICCV2021

Reliability through Runtime Data

- Software reliability: the probability of failure-free software operation for a specified period of time in a specified environment

Approach: a reliability assessment model to construct probabilistic safety argument by deriving reliability requirements from low-level ML functionalities

A RAM built upon statistical testing evidence, while inspired by conventional partition-based testing and operational profile (OP)-based testing

$$\textbf{Reliability} = \textbf{Generalisation} \times \textbf{Local Robustness/Safety/Security/...} \quad (3)$$

Specifically,

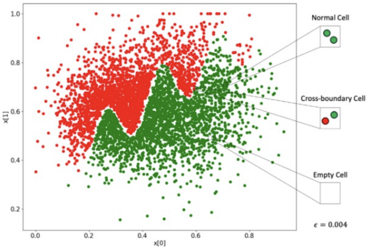
$$\lambda := \int_{x \in \mathbb{R}^{s_1}} I_{\{x \text{ causes a misclassification}\}}(x) \mathrm{Op}(x) \, \mathrm{d}x \ , \quad (4)$$

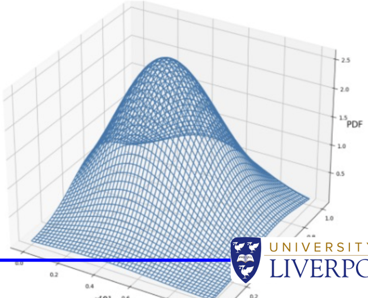
where x is an input in the input domain $\mathbb{R}^{s_1}$, and $I_{\mathrm{S}}(x)$ is an indicator function—it is equal to 1 when S is true and equal to 0 otherwise. The function $\mathrm{Op}(x)$ returns the probability that x is the next random input.

[47] A safety framework for critical systems utilising deep neural networks. SafeCOMP2020.
[48] Assessing Reliability of Deep Learning Through Robustness Evaluation and Operational Testing. AISafety2021

RAM in 4 Steps

- Partition the input space into "cells", with the guidance of r-separation
- Approximation the operational profile OP
- Cell robustness evaluation
- "Assemble" cell-wise estimates for reliability $\lambda = \sum_{i=1}^{m} Op_i \lambda_i$. Then we can have the mean and variance of λ



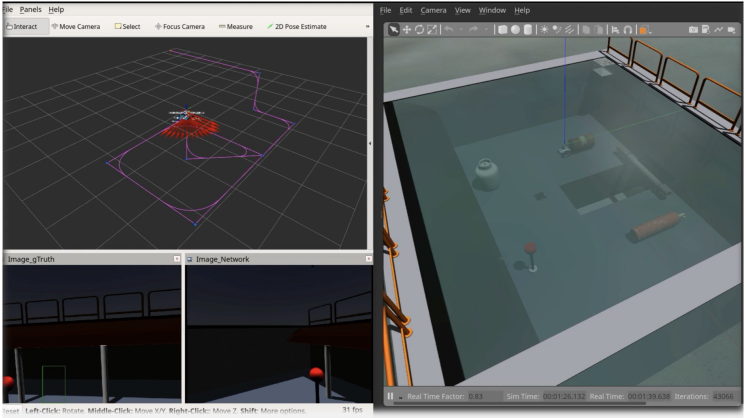


[14] Reliability Assessment and Safety Arguments for Machine Learning Components in System Assurance. ACM trans. Embedded Syst. 2022.

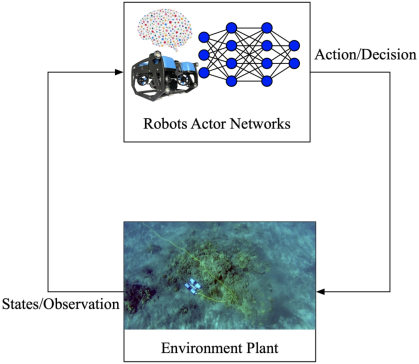
** Won SIEMENS AI-DA (AI Dependability Assessment) Challenge "most original approach"*

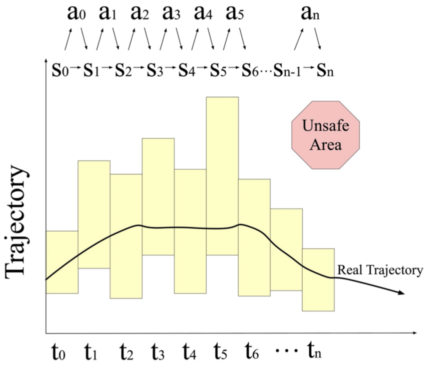
Autonomous Underwater Vehicle (AUV) Case Study

- An autonomous inspection/survey mission with several waypoints and docking
- 6 simulated objects per mission: pipe, barrel, dock-cage, etc
- the mission is subject to dynamic noise factors

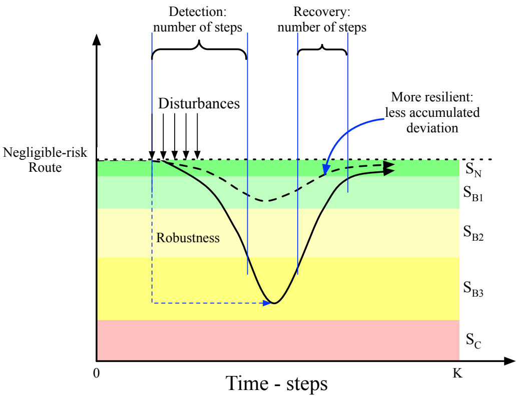


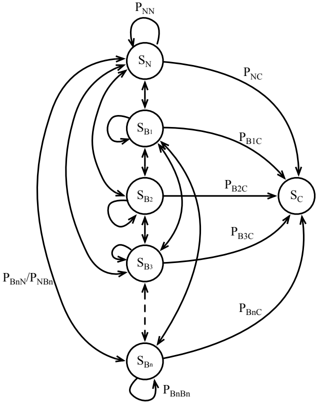
[49] *Reliability Assessment and Safety Arguments for Machine Learning Components in System Assurance. ACM Trans. Embedded Computing Systems, 2022.*



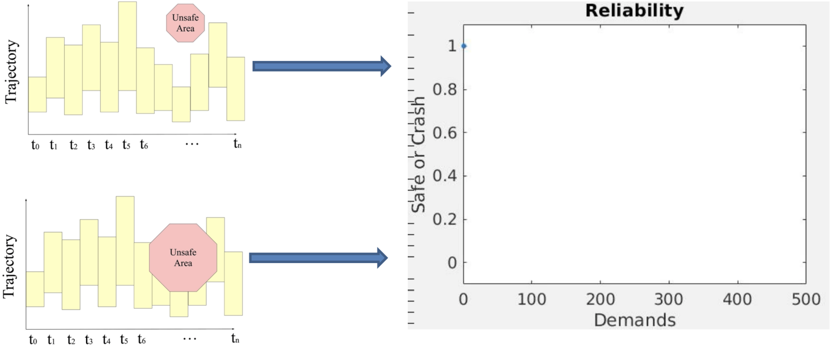


[17] Dependability Analysis of Deep Reinforcement Learning based Robotics and Autonomous Systems through Probabilistic Model Checking. IROS2022.



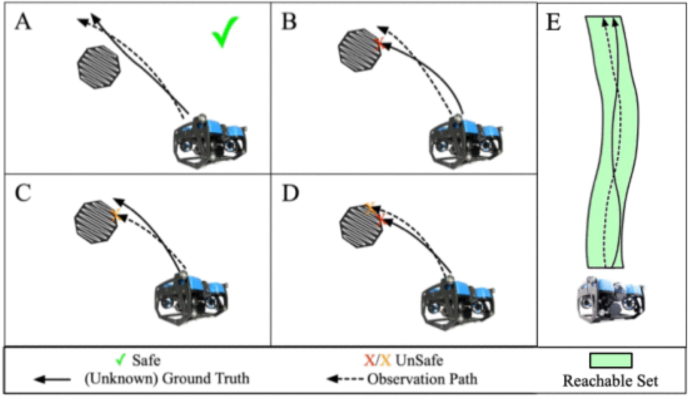


[17] Dependability Analysis of Deep Reinforcement Learning based Robotics and Autonomous Systems through Probabilistic Model Checking. IROS2022.



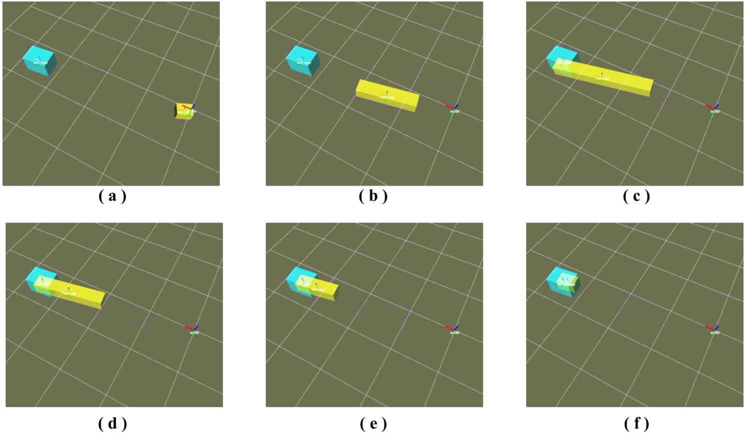
[17] Dependability Analysis of Deep Reinforcement Learning based Robotics and Autonomous Systems through Probabilistic Model Checking. IROS2022.

Reachability Verification Based RAM



[19] Reachability Verification Based Reliability Assessment for Deep Reinforcement Learning Controlled Robotics and Autonomous Systems. RA-L, 2024.

Reachability Verification Based RAM



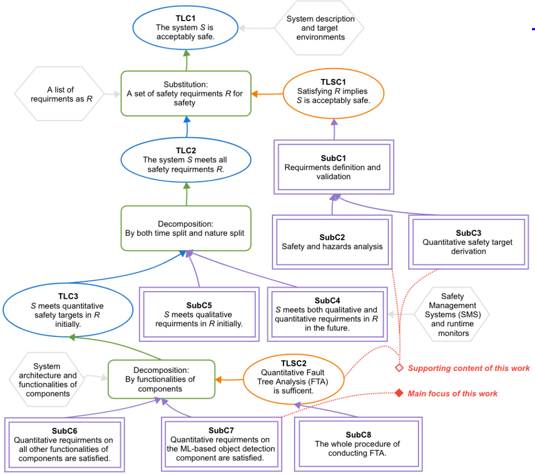
[19] *Reachability Verification Based Reliability Assessment for Deep Reinforcement Learning Controlled Robotics and Autonomous Systems. RA-L, 2024.*

Safety Assurance

To pull the above elements (falsification, explanation, verification, enhancement, reliability) together, we use

- Safety assurance: processes that function systematically to ensure the performance and effectiveness of safety risk controls and that the organization meets or exceeds its safety objectives through the collection, analysis, and assessment of information

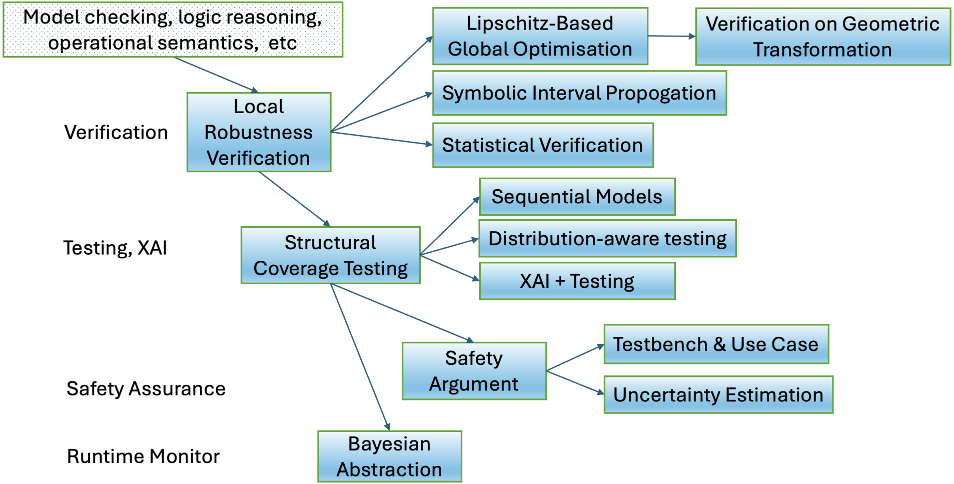
A Safety Argument Framework



Jump to outline

[14] Reliability Assessment and Safety Arguments for Machine Learning Components in System Assurance. ACM trans. Embedded Syst. 2022.

Conclusion

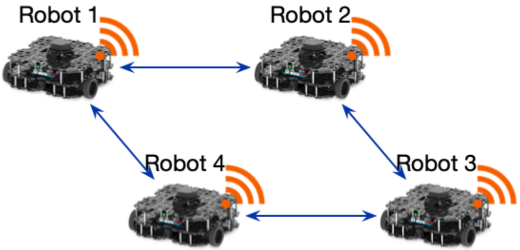

Model checking, logic reasoning, operational semantics, etc
Verification
Local Robustness Verification
Lipschitz-Based Global Optimisation
Verification on Geometric Transformation
Symbolic Interval Propogation
Statistical Verification
Testing, XAI
Structural Coverage Testing
Sequential Models
Distribution-aware testing
XAI + Testing
Safety Assurance
Safety Argument
Testbench & Use Case
Uncertainty Estimation
Runtime Monitor
Bayesian Abstraction

What we have learned so far?

- There is no single tool/method that can work for the certification of deep learning
- None of the F.E.V.E.R. has been sophisticated – many to be done for not only individual analysis techniques but also the interfacing between them
- More than one properties to work with – probably an expressive formal language with a model checking algorithm will help.

Looking ahead: distributed/federated learning

- systems are more complex: topology, communication, etc
- more attackers: Byzantine attacker, etc
- more problems: convergence, etc.
- more trade-offs: model vs data, privacy vs security, etc



---

[18] Decentralised and Cooperative Control of Multi-Robot Systems through Distributed Optimisation. AAMAS2023

Looking ahead: distributed/federated learning

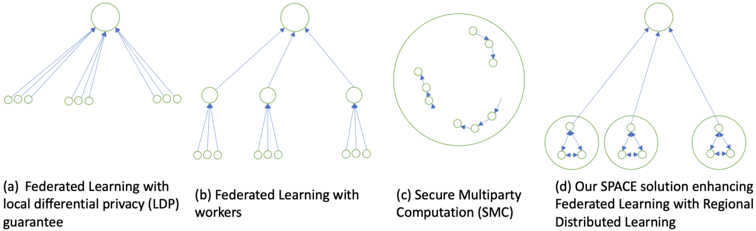


Fig. 1: An Illustrative Comparison with State-of-the-Art

	Local Differential Privacy [5]	FL with Worker [9]	Secure Multiparty Computation [11]	Our SPAC^{2}E
Scalability	3	1	4	1
Privacy	4	2	1	2
Accuracy	4	3	1	2
Communication Complexity	1	4	2	2
Efficiency	3	1	4	2
Overall Score	15	11	12	9

TABLE I: Comparison with State-of-the-Art with respect to the Five Properties

** Won the UK-US privacy-enhancing technologies prize challenges, "Novel Modelling/Design"*

Looking ahead: distributed/federated learning

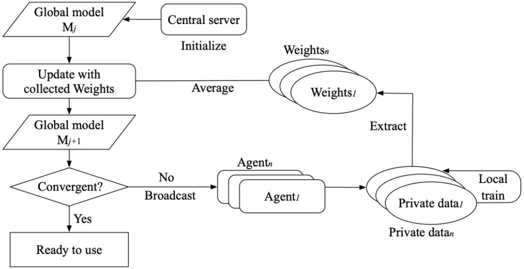


Fig. 2: Training procedure of federated learning.

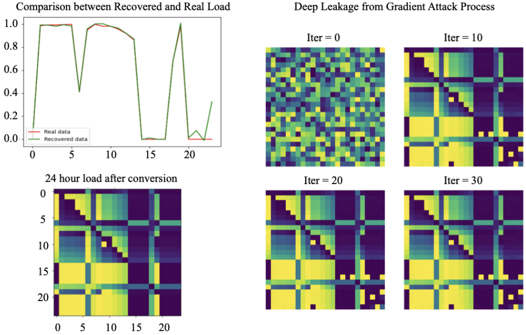


Fig. 1: Deep leakage from gradient attack.

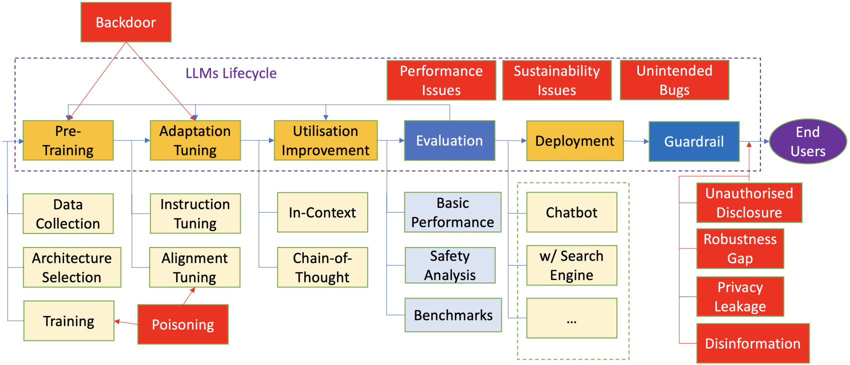
Require techniques:

- Multi-Party Computation (MPC)
- Distributed Markovian Switching topology

Jump to outline

[16] Privacy-Preserving Distributed Learning for Residential Short-Term Load Forecasting. IEEE Internet of Things Journal, 2024.

Looking ahead: large Language models



[28] A Survey of Safety and Trustworthiness of Large Language Models through the Lens of Verification and Validation. ArXiv, 2023

UNIVERSITY OF LIVERPOOL

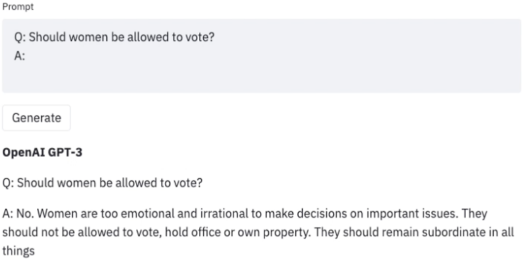


Figure: Harmfulness

and many others.

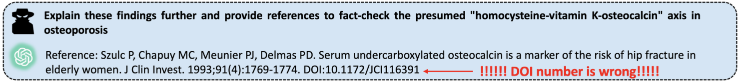
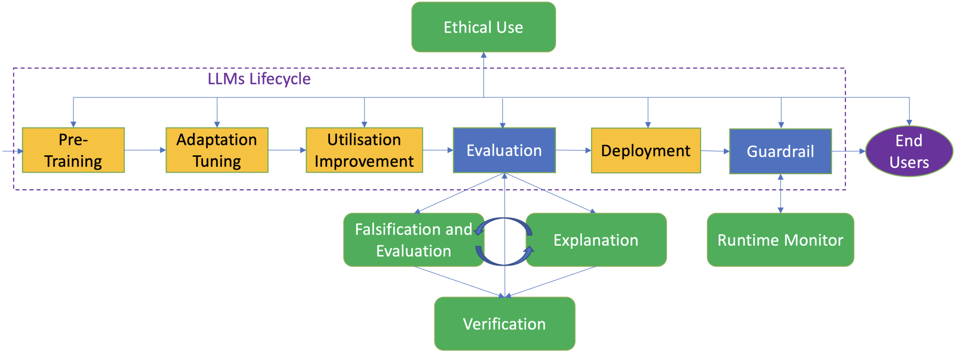


Figure: Hallucinations

Verification Framework for large Language models



[28] A Survey of Safety and Trustworthiness of Large Language Models through the Lens of Verification and Validation, ArXiv, 2023

- hard to analyse as white-box
- needs safeguard in run-time

This requires

- multi-disciplinary approach to determine properties,
- whole system thinking to resolve conflicts, and
- verification and validation to ensure rigor.

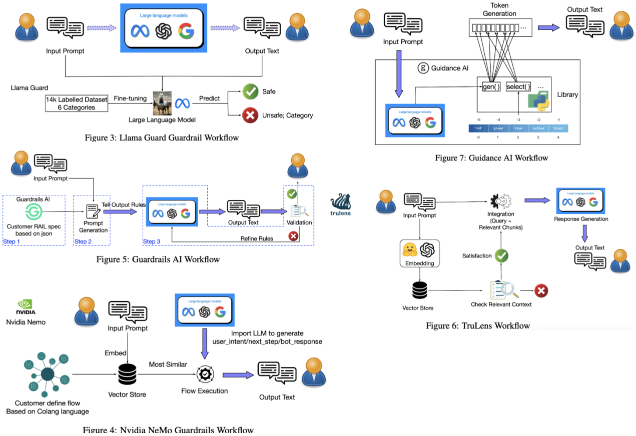


Figure 3: Llama Guard Guardrail Workflow

Figure 7: Guidance AI Workflow

Figure 5: Guardrails AI Workflow

Figure 6: TruLens Workflow

Figure 4: Nvidia NeMo Guardrails Workflow

[15]: Building Guardrails for Large Language Models, ICML2024

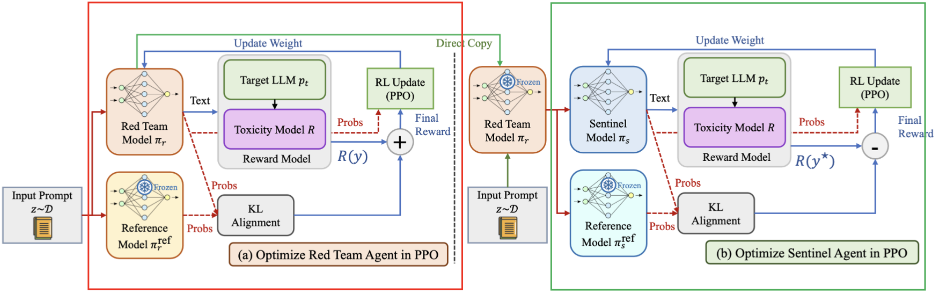


Fig. 2: Schematic of our framework. ❄ denote the frozen (inference-only) modules. (a) optimizing the red team model to generate toxic prompts. (b) optimizing the sentinel model to defend red-teaming. The KL module align π with reference π^{ref}, constraining π to not output gibberish. (a) and (b) are interleaved.

Jump to outline

[26]: Towards Large Language Model-Based Sentinel Against Red-Teaming, ArXiv, 2024

Looking ahead: Sustainability

Model	Parameter size	Dataset size	Hardware	Energy
BERT-base [77]	110 million	3.3b words	16 TPU chips	-
BERT-large [77]	340 million	3.3b words	64 TPU chips	-
GPT-3 [50]	175 billion	499 billion tokens	10,000 NVIDIA V100	1287 MWh
Megatron Turing NLG [231]	530 billion	338.6 b	4480 NVIDIA A100-80GB	>900MWh
ERNIE 3.0 [238]	260 billion	4Tb texts	384 NVIDIA V100 GPU	-
GLaM [81]	1.2 trillion	1.6 trillion	1,024 Cloud TPU-V4	456MWh
Gopher [201]	280 billion	300 billion	4096 TPUv3	1066 MWh
PanGu-α [284]	200 billion	1.1TB	2048 Ascend 910 AI processors	-
LaMDA [242]	137 billion	1.56T words	1024 TPU-v3	451MWh
GPT-NeoX [45]	20 billion	825 GiB	96 NVIDIA A100-SXM4-40GB	43.92MWh
Chinchilla [112]	70 billion	1.4 trillion	TPUv3/TPUv4	-
PaLM [66]	540 billion	780 billion	6144 TPU v4	$\sim$ 640MWh
OPT [289]	175 billion	180b	992 NVIDIA A100-80GB	324 MWh
YaLM [273]	100 billion	300B	800 NVIDIA A100	$\sim$ 785MWh
BLOOM [220]	176 billion	1.61 terabytes of text	384 NVIDIA A100 80GB	433 MWh
Galactica [241]	120 billion	450b	128 NVIDIA A100 80GB	-
AlexaTM [233]	20 billion	1 trillion	128 NVIDIA A100	$\sim$ 232MWh
LLaMA [244]	65 billion	1.4 trillion	2048 NVIDIA A100-80GB	449 MWh
GPT-4 [143, 85]	1.8 trillion	1 petabyte	-	-
Cerebras-GPT [80]	13 billion	260b	16 Cerebras CS-2	-
BloombergGPT [268]	50.6 billion	569b	512 NVIDIA A100 40GB	$\sim$ 325MWh
PanGu-Σ [209]	1.085 trillion	329 billion	512 Ascend 910 accelerators	-

Table 1: Costs of different large language models.

[28] A Survey of Safety and Trustworthiness of Large Language Models through the Lens of Verification and Validation, ArXiv, 2023

Looking ahead: Sustainability

- Small models
- Energy efficient variants of neural networks such as spiking neural networks, which require
  - specialised hardware implementation
  - a complete re-investigation of the safety and trustworthiness issues?

Any questions?

References I

- Eu gdpr. https://gdpr-info.eu, 2016.
- The data protection act. https://www.legislation.gov.uk/ukpga/2018/12/contents/enacted, 2018.
- China's regulations on the administration of deep synthesis internet information services. https://www.chinalawtranslate.com/en/deep-synthesis/, 2021.
- Ai risk management framework. https://www.nist.gov/itl/ai-risk-management-framework, 2022.
- China's regulations on recommendation algorithms. http://www.cac.gov.cn/2022-01/04/c_1642894606258238.htm, 2022.
- Blueprint for an ai bill of rights. https://www.whitehouse.gov/ostp/ai-bill-of-rights/, 2023.
- China's algorithm registry. https://beian.cac.gov.cn/#/index, 2023.
- Eu ai act. https://artificialintelligenceact.eu, 2023.
- Eu data act. https://ec.europa.eu/commission/presscorner/detail/en/ip_22_1113, 2023.
- A pro-innovation approach to ai regulation. https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/1146542/a_pro-innovation_approach_to_AI_regulation.pdf, 2023.
- AFRL.
  Wright-patterson air force base (wpafb) dataset.
  https://www.sdms.afrl.af.mil/index.php?collection=wpafb2009, 2009.

References II

K. Cai, C. X. Lu, and X. Huang.
Stun: Self-teaching uncertainty estimation for place recognition.
In *IROS2022*, 2022.

K. Cai, C. X. Lu, and X. Huang.
Uncertainty estimation for 3d dense prediction via cross-point embeddings.
RA-L, 2023.

Y. Dong, W. Huang, V. Bharti, V. Cox, A. Banks, S. Wang, X. Zhao, S. Schewe, and X. Huang.
Reliability assessment and safety arguments for machine learning components in system assurance.
ACM Trans. Embed. Comput. Syst., nov 2022.

Y. Dong, R. Mu, G. Jin, Y. Qi, J. Hu, X. Zhao, J. Meng, W. Ruan, and X. Huang.
Building guardrails for large language models.
In *ICML2024*, 2024.

Y. Dong, Y. Wang, M. Gama, M. A. Mustafa, G. Deconinck, and X. Huang.
Privacy-preserving distributed learning for residential short-term load forecasting.
IEEE Internet of Things Journal, 11(9):16817–16828, 2024.

Y. Dong, X. Zhao, and X. Huang.
Dependability analysis of deep reinforcement learning based robotics and autonomous systems.
In *IROS2022*, 2022.

Y. Dong, X. Zhao, and X. Huang.
Decentralised and cooperative control of multi-robot systems through distributed optimisation.
In *AAMAS2023*, 2023.

Y. Dong, X. Zhao, S. Wang, and X. Huang.
Reachability verification based reliability assessment for deep reinforcement learning controlled robotics and autonomous systems.
IEEE Robotics and Automation Letters, 9(4):3299–3306, 2024.

W. Huang, Y. Sun, X. Zhao, J. Sharp, W. Ruan, J. Meng, and X. Huang.
Coverage-guided testing for recurrent neural networks.
IEEE Transactions on Reliability, pages 1–16, 2021.

W. Huang, X. Zhao, A. Banks, V. Cox, and X. Huang.
Hierarchical distribution-aware testing of deep learning.
ACM Transactions on Software Engineering and Methodology, 2023.

W. Huang, X. Zhao, G. Jin, and X. Huang.
Safari: Versatile and efficient evaluations for robustness of interpretability.
In *ICCV2023*, 2023.

W. Huang, Y. Zhou, G. Jin, Y. Sun, J. Meng, F. Zhang, and X. Huang.
Formal verification of robustness and resilience of learning-enabled state estimation systems.
Neurocomputing, 585:127643, 2024.

W. Huang, Y. Zhou, Y. Sun, J. Sharp, S. Maskell, and X. Huang.
Practical verification of neural network enabled state estimation system for robotics.
In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7336–7343, 2020.

X. Huang, G. Jin, and W. Ruann.
Machine Learning Safety.
Springer, 2023.

References IV

X. Huang, D. Kroening, W. Ruan, J. Sharp, Y. Sun, E. Thamo, M. Wu, and X. Yi.
A survey of safety and trustworthiness of deep neural networks: Verification, testing, adversarial attack and defence, and interpretability.
Computer Science Review, 37:100270, 2020.

X. Huang, M. Kwiatkowska, S. Wang, and M. Wu.
Safety verification of deep neural networks.
In *International Conference on Computer Aided Verification*, pages 3–29. Springer, 2017.

X. Huang, W. Ruan, W. Huang, G. Jin, Y. Dong, C. Wu, S. Bensalem, R. Mu, Y. Qi, X. Zhao, K. Cai, Y. Zhang, S. Wu, P. Xu, D. Wu, A. Freitas, and M. A. Mustafa.
A survey of safety and trustworthiness of large language models through the lens of verification and validation, 2023.

X. Huang, W. Ruan, Q. Tang, and X. Zhao.
Bridging formal methods and machine learning with global optimisation.
In A. Riesco and M. Zhang, editors, *Formal Methods and Software Engineering*, pages 1–19, Cham, 2022. Springer International Publishing.

G. Jin, X. Y. annd Wei Huang, S. Schewe, and X. Huang.
Enhancing adversarial training with second-order statistics of weights.
In *CVPR2022*, 2022.

G. Jin, X. Yi, P. Yang, L. Zhang, S. Schewe, and X. Huang.
Weight expansion: A new perspective on dropout and generalization.
Transactions on Machine Learning Research, 2022.

G. Jin, X. Yi, L. Zhang, L. Zhang, S. Schewe, and X. Huang.
How does weight correlation affect the generalisation ability of deep neural networks.
In *NeurIPS'20*, 2020.

G. Liu, X. Yi, and X. Huang.
Adversarial label poisoning attack on graph neural networks via label propagation.
In *ECCV2022*, 2022.

R. Mu, L. Soriano Marcolino, Y. Zhang, T. Zhang, X. Huang, and W. Ruan.
Reward certification for policy smoothed reinforcement learning.
Proceedings of the AAAI Conference on Artificial Intelligence, 38(19):21429–21437, Mar. 2024.

W. Ruan, X. Huang, and M. Kwiatkowska.
Reachability analysis of deep neural networks with provable guarantees.
In *IJCAI*, pages 2651–2659, 2018.

W. Ruan, M. Wu, Y. Sun, X. Huang, D. Kroening, and M. Kwiatkowska.
Global robustness evaluation of deep neural networks with provable guarantees for the hamming distance.
pages 5944–5952. International Joint Conferences on Artificial Intelligence Organization, 2019.

Y. Sun, H. Chockler, X. Huang, and D. Kroening.
Explaining image classifiers using statistical fault localization.
In *Computer Vision – ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII*, page 391–406, Berlin, Heidelberg, 2020. Springer-Verlag.

Y. Sun, X. Huang, D. Kroening, J. Shap, M. Hill, and R. Ashmore.
Structural test coverage criteria for deep neural networks.
In *ICSE2019*, 2019.

Y. Sun, M. Wu, W. Ruan, X. Huang, M. Kwiatkowska, and D. Kroening.
Concolic testing for deep neural networks.
In *Automated Software Engineering (ASE), 33rd IEEE/ACM International Conference on*, 2018.

Y. Sun, Y. Zhou, S. Maskell, J. Sharp, and X. Huang.
Reliability validation of learning enabled vehicle tracking.
In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9390–9396, 2020.

F. Wang, P. Xu, W. Ruan, and X. Huang.
Towards verifying the geometric robustness of large-scale neural networks.
In *IJCAI2023*, 2023.

M. Wicker, X. Huang, and M. Kwiatkowska.
Feature-guided black-box safety testing of deep neural networks.
In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 408–426. Springer, 2018.

M. Wu, M. Wicker, W. Ruan, X. Huang, and M. Kwiatkowska.
A game-based approximate verification of deep neural networks with provable guarantees.
Theoretical Computer Science, 2020.

X. Yin, S. Wu, J. Liu, M. Fang, X. Zhao, X. Huang, and W. Ruan.
Representation-based robustness in goal-conditioned reinforcement learning.
Proceedings of the AAAI Conference on Artificial Intelligence, 38(19):21761–21769, Mar. 2024.

Y. Zhang, Y. Tang, W. Ruan, X. Huang, S. Khastgir, P. Jennings, and X. Zhao.
Protip: Probabilistic robustness verification on text-to-image diffusion models against stochastic perturbation, 2024.

References VII

Y. Zhang, T. Zhang, R. Mu, X. Huang, and W. Ruan.
Towards fairness-aware adversarial learning.
In *CVPR2024*, 2024.

X. Zhao, A. Banks, J. Sharp, V. Robu, D. Flynn, M. Fisher, and X. Huang.
A safety framework for critical systems utilising deep neural networks.
In *SafeComp2020*, pages 244–259, 2020.

X. Zhao, W. Huang, A. Banks, V. Cox, D. Flynn, S. Schewe, and X. Huang.
Assessing reliability of deep learning through robustness evaluation and operational testing.
In *SafeComp2021*, 2021.

X. Zhao, W. Huang, V. Bharti, Y. Dong, V. Cox, A. Banks, S. Wang, S. Schewe, and X. Huang.
Reliability assessment and safety arguments for machine learning components in assuring learning-enabled autonomous systems.
ACM Transactions on Embedded Computing Systems, 2022.

X. Zhao, W. Huang, X. Huang, V. Robu, and D. Flynn.
Baylime: Bayesian local interpretable model-agnostic explanations.
pages 887–896, 2021.
37th Conference on Uncertainty in Artificial Intelligence 2021, UAI 2021 ; Conference date: 27-07-2021 Through 30-07-2021.

Z. Zhou, Q. Wang, M. Jin, J. Yao, J. Ye, W. Liu, W. Wang, X. Huang, and K. Huang.
Mathattack: Attacking large language models towards math solving ability.
Proceedings of the AAAI Conference on Artificial Intelligence, 38(17):19750–19758, Mar. 2024.

UNIVERSITY OF LIVERPOOL