responsible AI

DLA - advance deep learning 2026

CTU in Prague

Giorgos Tolias

responsible AI

- responsible AI is the practice of developing and deploying artificial intelligence systems that are ethical, safe, transparent, and accountable, ensuring they align with human values and societal benefits

- fairness and non-discrimination
- explainability and transparency
- security and robustness
- privacy and data protection
- intellectual property and ownership
- environmental and social impact
- accountability

fairness

fairness and why it matters

- a model's decisions and errors should not systematically disadvantage some groups compared to others
  - fairness is about who benefits, who is harmed, and who gets more errors

- an accurate model can still be unfair if its errors are distributed unevenly across groups, eg:
  - face recognition worse on some skin tones or genders
  - hiring or admissions models inheriting historical bias
  - medical risk models underperforming on underrepresented populations

where unfairness comes from

- data → labels → model → deployment

- data bias: some groups underrepresented
- label bias: labels reflect human prejudice or noisy proxies
- objective bias: optimizing average accuracy hides subgroup failures
- deployment bias: model used in a setting different from training

- models do not invent social patterns from nowhere, they often amplify what is already in the data and objective

Tech

Amazon Pulled the Plug on an AI Recruitment Tool That Was Biased Against Women

By Samantha Cole October 10, 2018, 11:49am

slide credit: Noa Garcia

	TYPE I	TYPE II	TYPE III	TYPE IV	TYPE V	TYPE VI
Microsoft	1.7%	1.1%	3.3%	0%	23.2%	25.0%
IBM	5.1%	7.4%	8.2%	8.3%	33.3%	**46.8%**
FACE++	11.9%	9.7%	8.2%	13.9%	32.4%	**46.5%**

J. Buolamwini and T. Gebru (2018). Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification.

slide credit: Noa Garcia

socio-technical problem

- fairness in ML is a socio-technical problem
  - it involves human values, but also data, modeling choices, and measurable disparities

- mitigation via constraints during training
  - equalizing loss, acceptance rate, false positive/negative error across groups

- technical notation
  - X: input
  - $Y \in \{0,1\}$: true label
  - $A \in G$: group
  - $s_\theta(X) \in [0,1]$: model score
  - $\hat{Y} = \mathbf{1}[s_\theta(X) \geq t] \in \{0,1\}$: predicted label

equal predictive quality (loss)

- the model should perform equally well across groups

- minimize loss averaged over groups

$$\min_{\theta} \frac{1}{|G|} \sum_{g \in G} \mathbb{E}[\ell(s_\theta(X), Y) \,|\, A = g]$$

- fairness-aware objective
  - each group contributes equally to training, useful when some groups are underrepresented

- does not directly enforce equal acceptance rates or equal error rates
  - depends on the chosen task loss

equal acceptance rate

- the model should predict the positive class equally often across groups

- hard constraint - non-differentiable

$$P(\hat{Y} = 1 \,|\, A = g) = P(\hat{Y} = 1 \,|\, A = g'), \quad g, g' \in G$$

- soft constraint - as regularization in training
  - soft surrogate uses average score instead of hard thresholded decisions

$$\Big(\mathbb{E}\left(s_\theta(X) \,|\, A = g\right) - \mathbb{E}\left(s_\theta(X) \,|\, A = g'\right) \Big)^2$$

- ignores the true label
- parity of outcomes not of mistakes
  - same acceptance rate does not mean same error rate

equal mistake profile (error rates)

- the model should make the same kinds of mistakes across groups
- equal false-positive and false-negative rates

- regularization via soft surrogate of the hard constraint

$$\mathrm{FPR}_g = \mathrm{FPR}_g^{'} \quad \text{and} \quad \mathrm{FNR}_g = \mathrm{FNR}_g^{'}$$

- conditions on the true label
- equal treatment of qualified/unqualified individuals
- does not guarantee equal acceptance rate

explainability

don't blindly trust the model

- a deep model predicting COVID relied on hospital watermarks on the x-ray (DeGrave 2020)
- a machine learning model trained to detect whales learned to rely on artifacts in audio files instead of basing the classification on the audio content (DeLMA and Cukierski 2013)
- a wolf versus dog classifier relied on snow in the background instead of image regions that showed the animals (Ribeiro, Singh, and Guestrin 2016)

- sensitive domains: health care, justice, insurance, ..., chat LLMs

explainability / interpretability

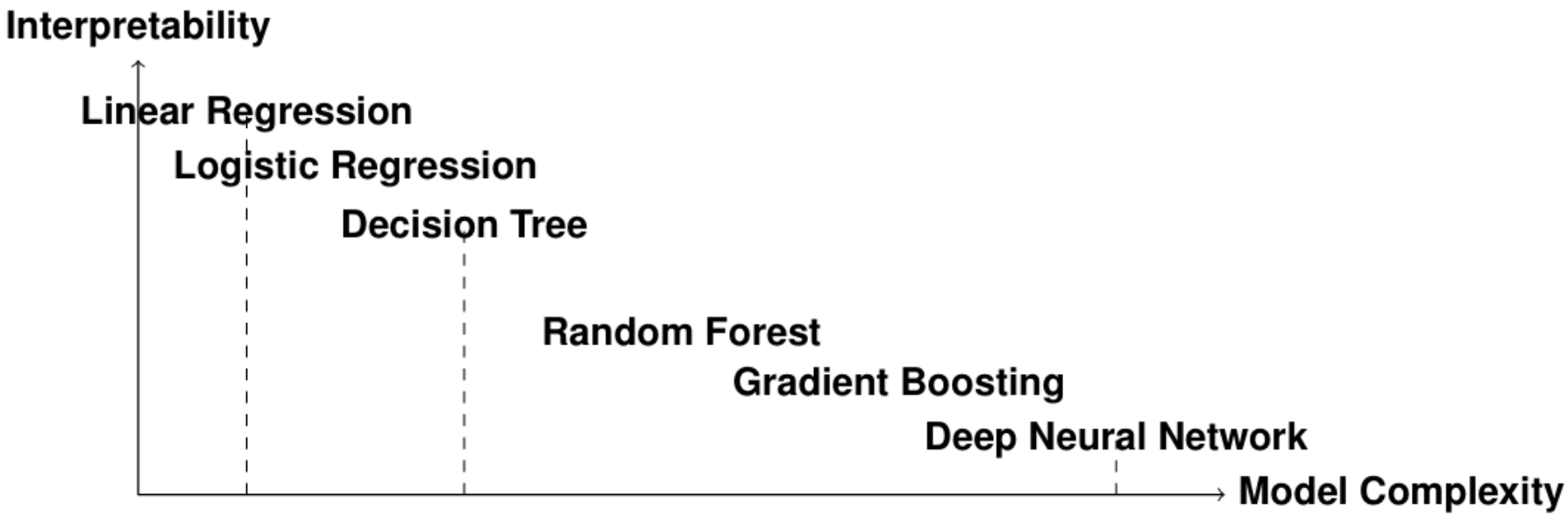
- ability to understand
  - how the model makes decisions?
  - why the model produces a specific output?

- essential for
  - debugging tool
  - social acceptance
  - human machine interaction

tough to evaluate

- lack of ground truth for explanations
  - exact reasoning behind the model's decision is unknown
- subjectivity and human involvement
  - context-specific and user-dependent

- lack of concrete definition
- not just a technical challenge - also a philosophical problem

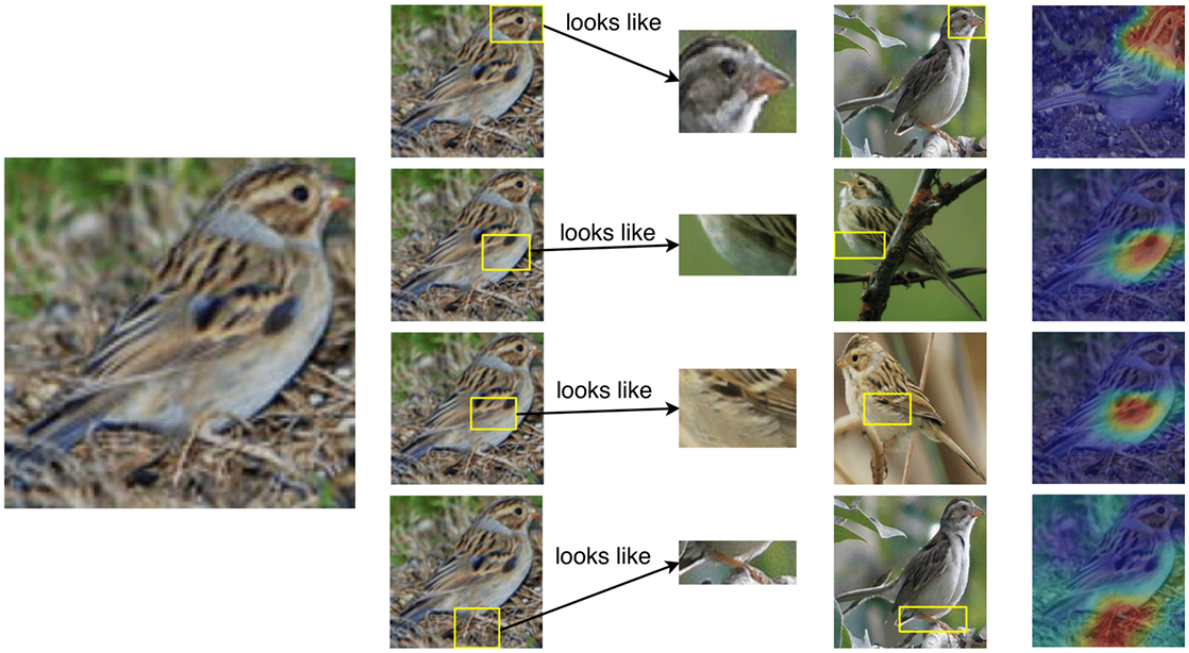

Interpretability
Linear Regression
Logistic Regression
Decision Tree
Random Forest
Gradient Boosting
Deep Neural Network
Model Complexity

taxonomy of ML explainability methods

- by-design explainability
  - the model is built to be understandable from the start
- post-hoc explainability
  - the model is already trained, often complex, and we explain it afterward
- model-agnostic
  - treat the model as a black box - only use inputs and outputs
- model-specific
  - exploit the internal structure of a particular model family
- local
  - explain one specific prediction
- global
  - explain the model's general behavior

- which parts of the input are responsible for a prediction?
- which training examples are responsible for a prediction?

nearest neighbors - prototypical parts

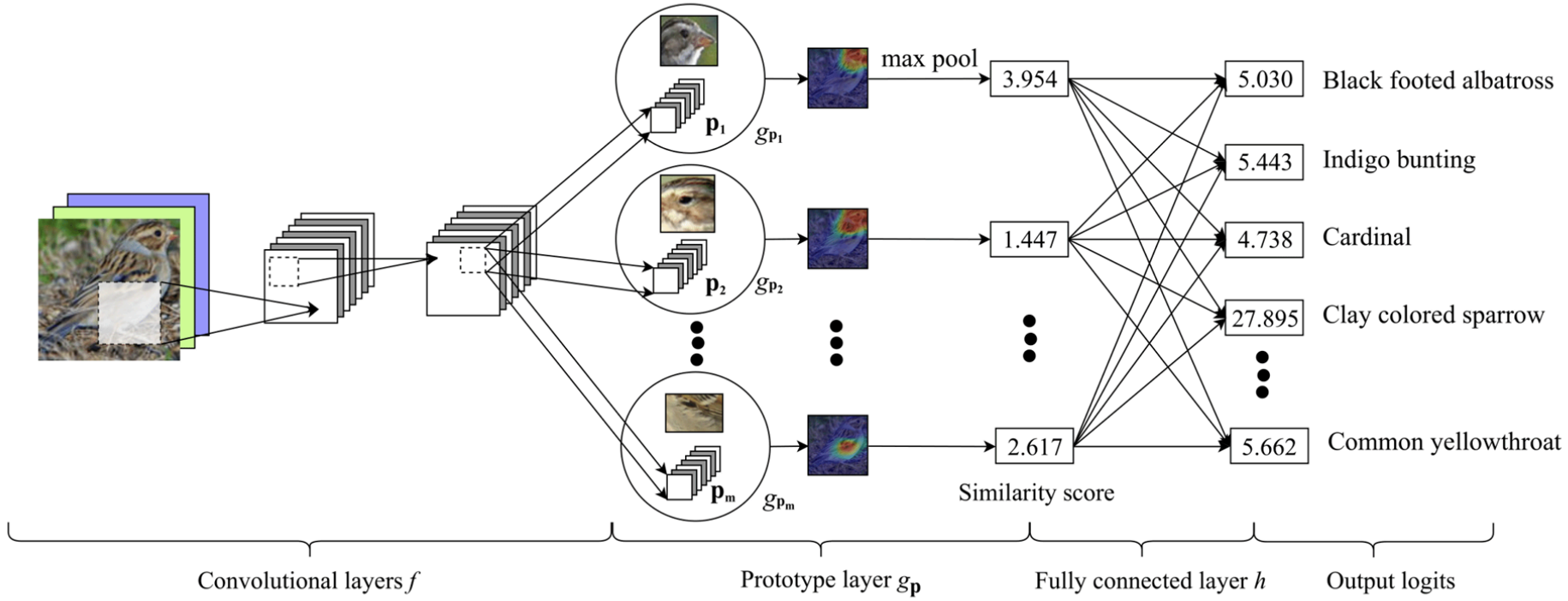


Leftmost: a test image of a clay-colored sparrow
Second column: same test image, each with a bounding box generated by our model -- the content within the bounding box is considered by our model to look similar to the prototypical part (same row, third column) learned by our algorithm
Third column: prototypical parts learned by our algorithm
Fourth column: source images of the prototypical parts in the third column
Rightmost column: activation maps indicating how similar each prototypical part resembles part of the test bird

- the model (eg. its features) is not interpretable but the prediction is
- nearest neigbor models are interpretable

[Chen, NeurIPS19]

nearest neighbors - prototypical parts



- image passed through convolutional layers: obtain a spatial feature map, each spatial location corresponds to an image patch
- each prototype vector corresponds to a training image patch
- compare each prototype against all patches in the feature map, keep the strongest match
- each class logit is a weighted sum of prototype activations

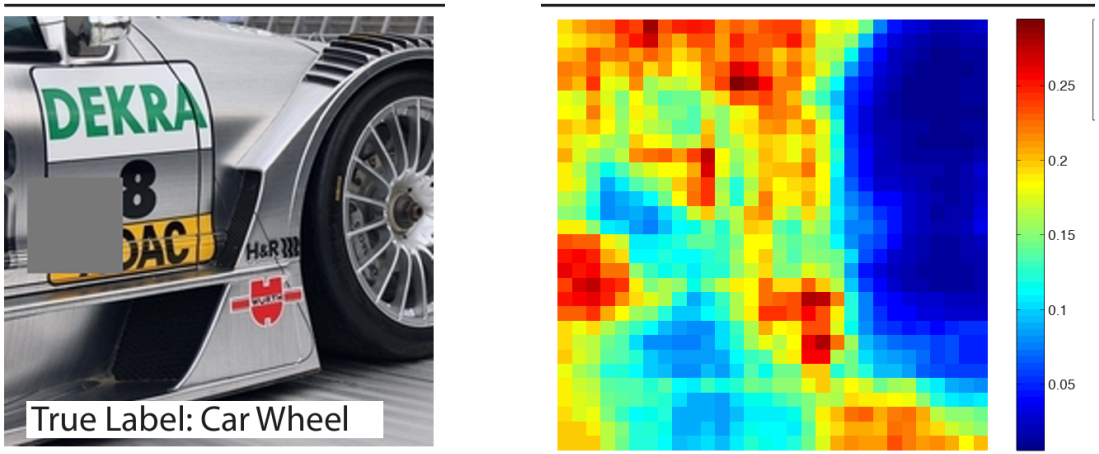
[Chen, NeurIPS19]

explainability maps

- which part of the input is important?

[Simonyan, ICLR14]

pixel gradients $|\frac{\partial y_c}{\partial x_{ij}}|$



[Zeiler, ECCV14]

influence via masking $|f(x) - f(x \odot m)|$

explainability maps

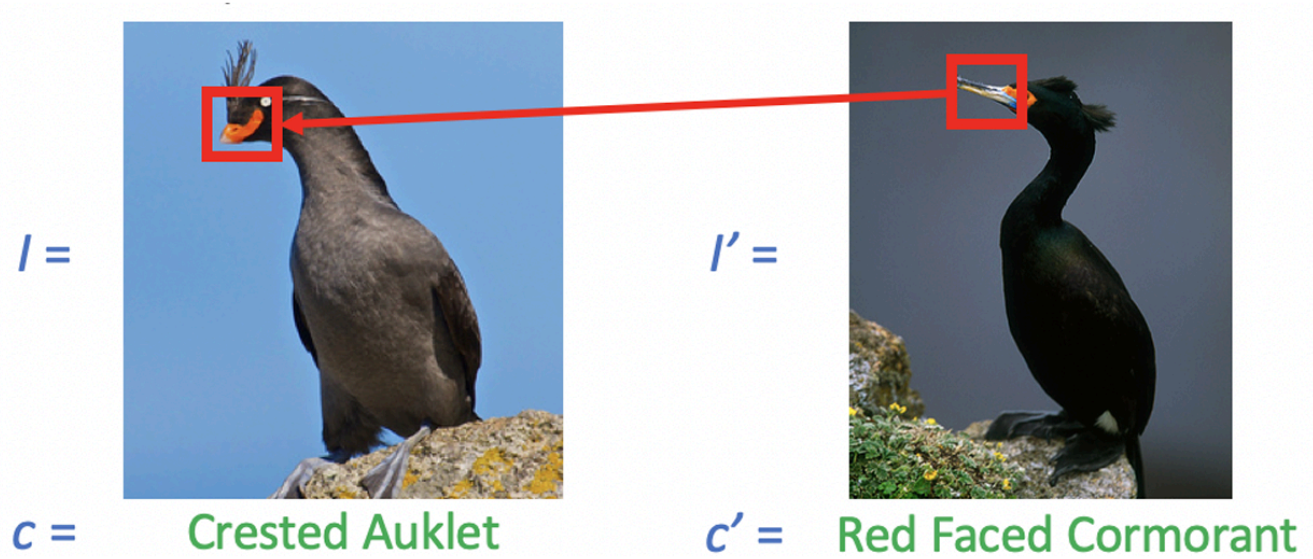
- which part of the input is important?

- last conv. feature map: compute gradients of the class score
- weight feature maps by importance $w_k = \sum_{ij} \text{ReLU}(\frac{\partial y_c}{\partial A^k_{ij}})$
- combine the feature maps: the weighted feature maps are summed into one coarse heatmap

[Zhou, CVPR16] [Selvaraju, ICCV17]

counterfactual explanations

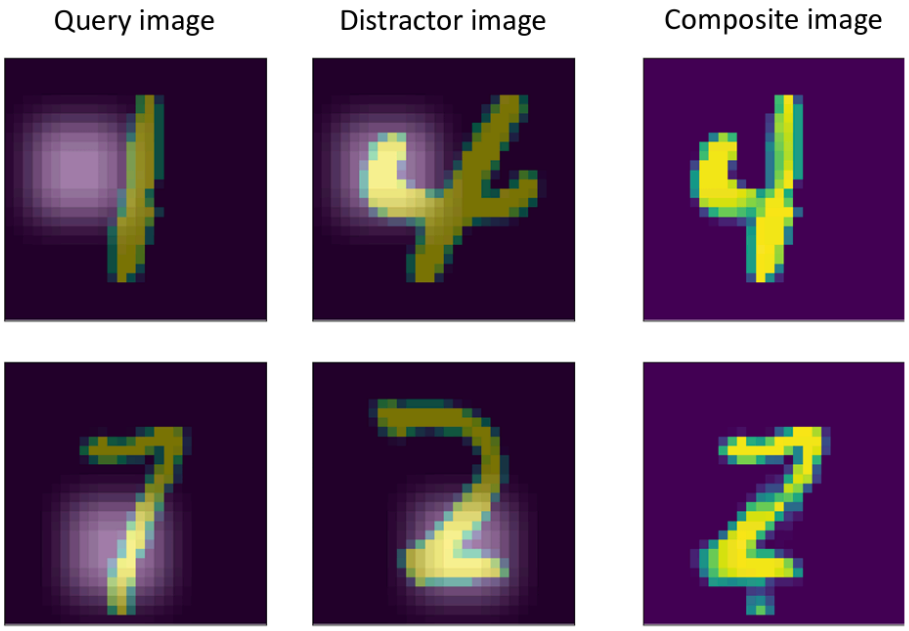
- what minimal change would make the model change its prediction from one class to another
- change: swap image patches (in the feature space)
- keep most patch features from the original image, but replace selected features by features from the target image

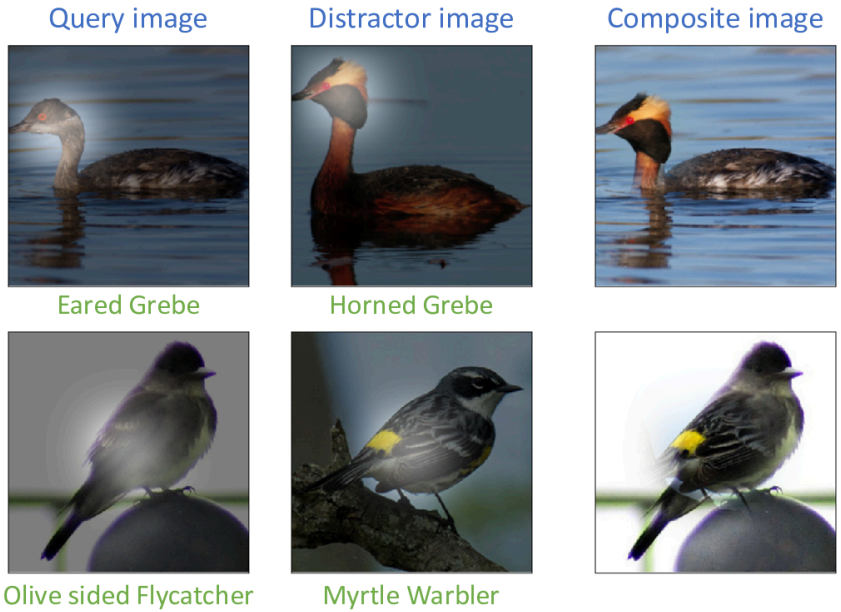


[Goyal, PMLR19]

counterfactual explanations

- this is a counterfactual in representation space, not a realistic image
  - it tells us which internal features would need to change





[Goyal, PMLR19]

869 → 85

979 → 197

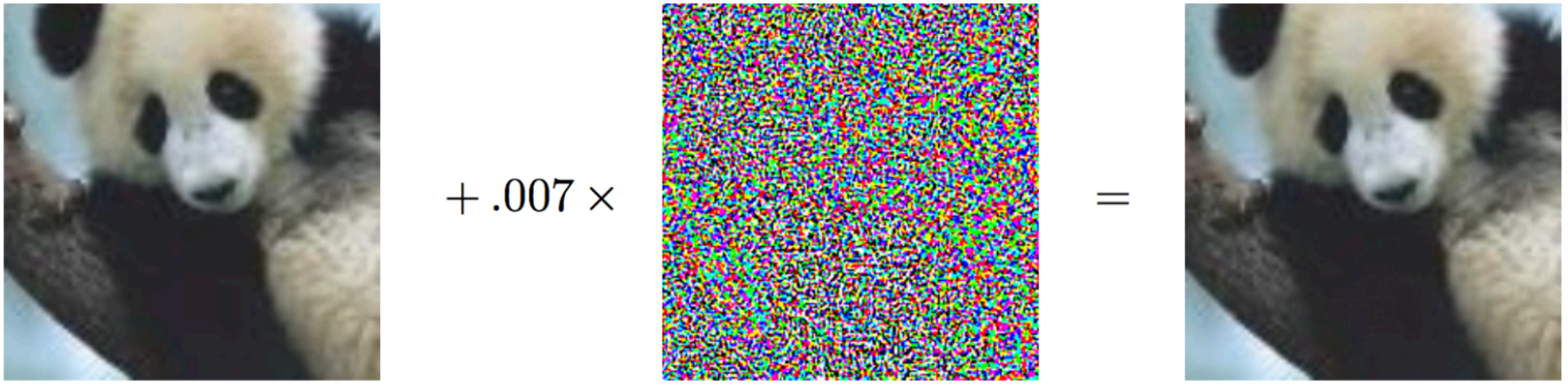
649 → 7

[Suma, ECCV24]

security

adversarial attacks

- ML systems can be manipulated on purpose
- goal: cause a wrong prediction while keeping the change small
- key difference from ordinary errors: the input is chosen by an attacker
- security relevance: ML can fail under deliberate manipulation
- example: digital content moderation, traffic sign manipulation



fast gradient sign method (FGSM)

- add a small perturbation to the input image

$$x' = x + \delta, \qquad \|\delta\|_p \leq \varepsilon$$

- optimize perturbation to increase the loss

$$\max_{\|\delta\|_p \leq \varepsilon} L\big(f(x+\delta), y\big)$$

- FGSM: a simple gradient-based single-step attack

$$x' = x + \varepsilon \operatorname{sign}\big(\nabla_x L\big(f(x), y\big)\big)$$

  - specifically designed for L_∞ constraint

- even a tiny perturbation can change the prediction

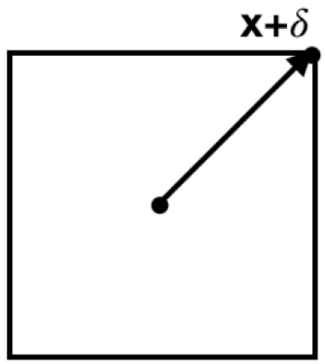
projected gradient descent

- iterative attack: multiple small gradient steps
- after each step, project back to the allowed perturbation set

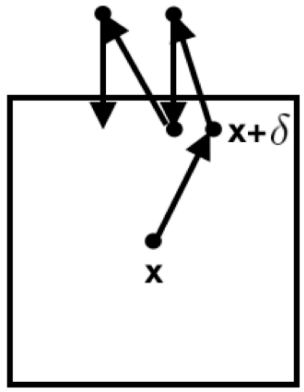
$$x^{(t+1)} = \Pi_S\left(x^{(t)} + \alpha \nabla_x L\left(f(x^{(t)}), y\right)\right)$$

- projection enforces the constraint

$$\Pi_S(z) = \arg \min_{u:\|u-x\|_p \leq \varepsilon} \|u - z\|_2.$$



(a) Fast Gradient Sign Method (FGSM)



(b) Projected Gradient Descent (PGD)

figure credit: A. Mądry

(non) targetted attacks and thread model

- non-targeted attack: any wrong prediction is sufficient

$$\max_{\|\delta\|_p \leq \varepsilon} L\big(f(x+\delta), y\big)$$

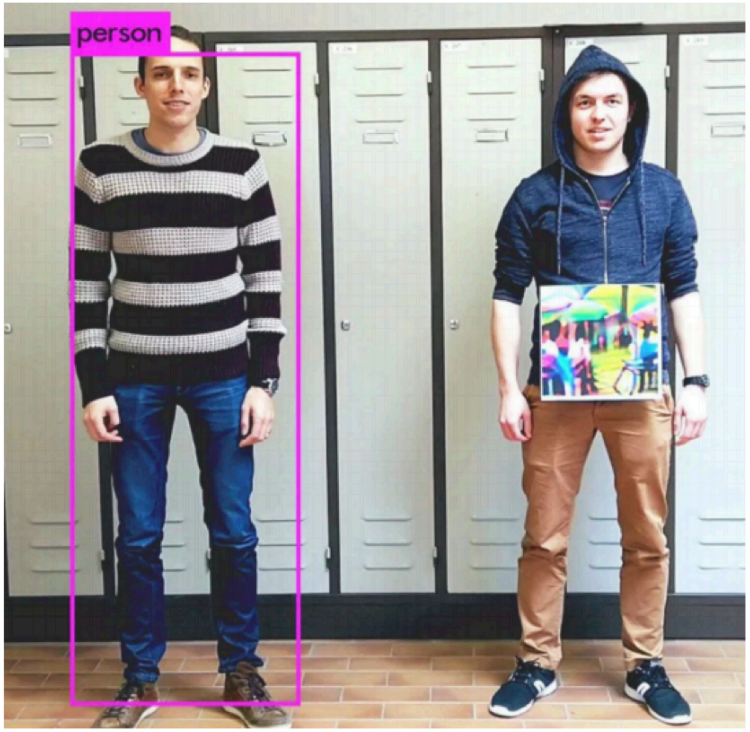
- targeted attack: force the model toward a specific target class

$$\min_{\|\delta\|_p \leq \varepsilon} L\big(f(x+\delta), y'\big)$$

- white-box: knows model, parameters, gradients
- black-box: only queries or only observes outputs
  - black-box via surrogate model: train a substitute model from query access, then craft attacks on the substitute
- transferability: adversarial examples crafted on one model can often fool another

physical world attacks

- not every digital attack is practical, but some physical attacks are



Adversarial “T-shirt”
(Xu et al 2020)

Robust Physical-World Attacks on Deep Learning Visual Classification, CVPR’18

defense - adversarial training

- train on adversarially perturbed examples

$$\min_{\theta} \; \mathbb{E}_{(x,y)\sim D} \left[\max_{\|\delta\|_p \leq \varepsilon} L\big(f_\theta(x+\delta), y\big) \right]$$

- improves local robustness within the chosen perturbation model
- adversarial training encourages locally flatter loss landscapes and more stable predictions around training examples
- trade-off: higher training cost, often lower clean accuracy

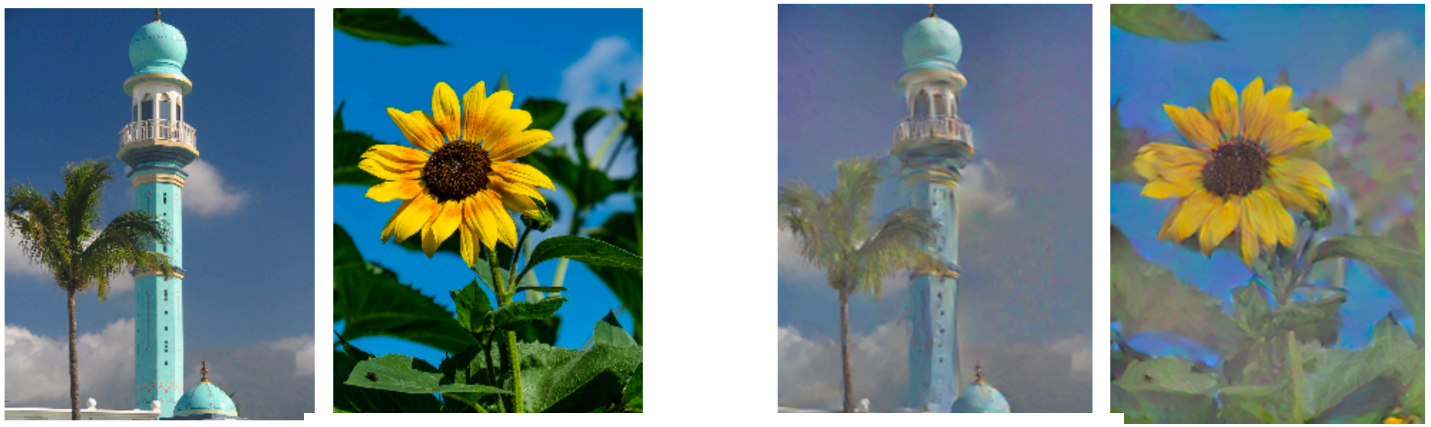
privacy

model inversion attacks - reconstruct sensitive features

- can we reconstruct the input given the model's output?
- classical computer vision system representing an image with a set of local descriptors [Weinzaepfel, CVPR'11]

model inversion attacks - reconstruct sensitive features

- can we reconstruct the input given the model's output?
  - for model $y = f_\theta(x)$, inversion by $x = f_\phi^{-1}(y)$
- optimization-based inversion: $\underset{\hat{x}}{\text{argmin}}\ L(f_\theta(\hat{x}), y) + \lambda R(\hat{x})$



[Tolias, ICCV'19]

[Fredrikson, CSS'15]

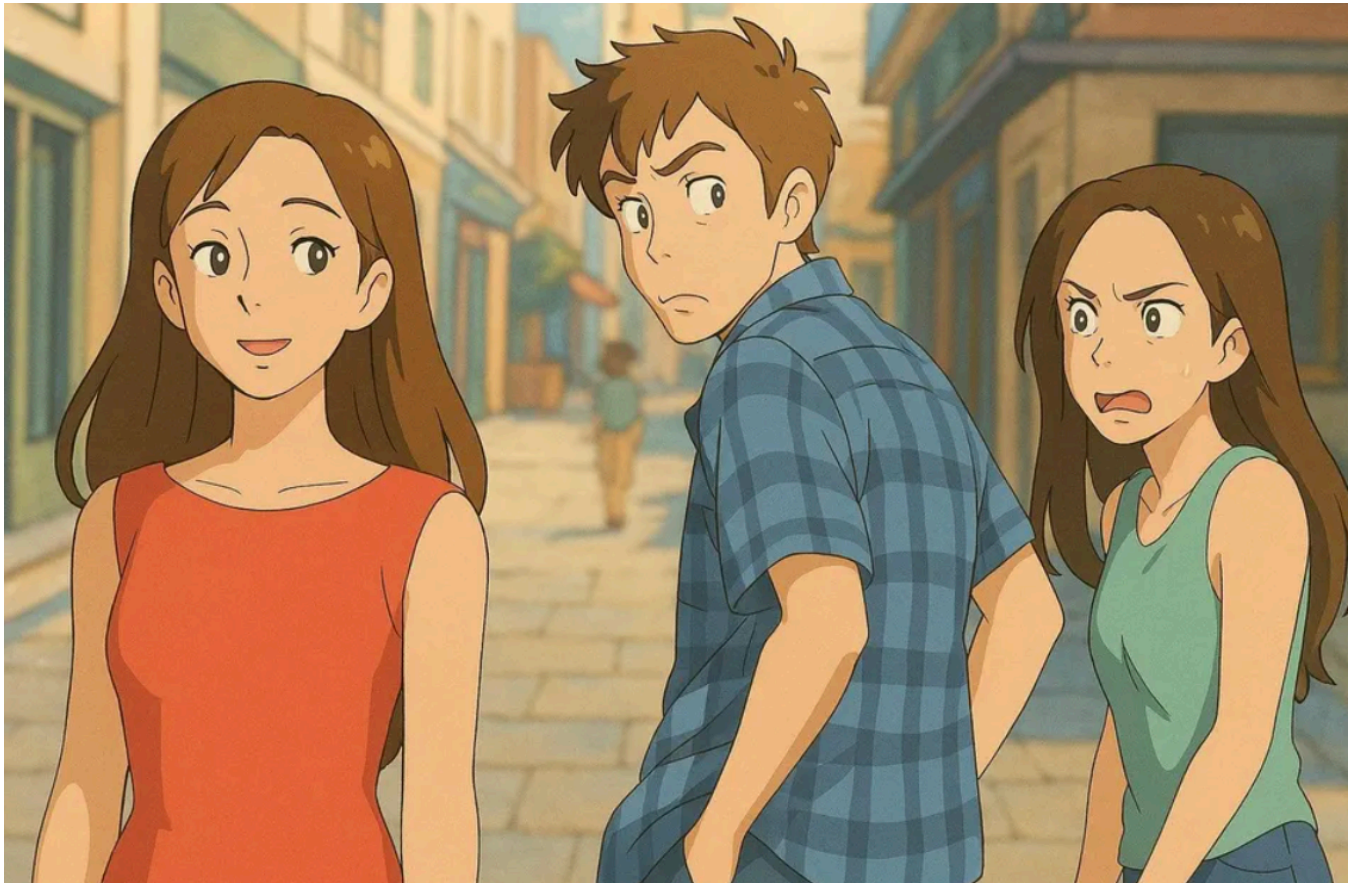
defense against model inversion attacks

- reduce memorization / overfitting
  - overfitting and overconfident models leak more
  - dropout, weight decay, data augmentation, early stopping
- differential privacy during training
  - clip gradients and add noise — formal privacy guarantees
  - usually trades off some accuracy
- system-level defenses - limit information in the output
  - query rate limiting
  - return only top-1 label or coarse scores
  - avoid full confidence vectors or fine-grained logits
  - confidence clipping, rounding

intellectual property

can a style be copyrighted?

- public discussion: online generative models procuding images in the style of Studio Ghibli



the richest source
of public domain images

WIKIMEDIA
COMMONS

120M public domain images

common
training dataset

LAION

5B text-image pairs


ChatGPT

?


deepseek

watermarking of images

- invisible whatevermarks of copyrights images
  - predict whether such images were used in model training (classification task)
- radioactive image: add invisible watermark
  - s.t. $\phi(x)$ shifts by $\alpha u, \alpha \in \mathbb{R}, u \in \mathbb{R}^d$,
  - assume a known feature extractor ϕ
- white-box — access to the classifier
  - large cosine sim. between classifier vector and $u \rightarrow$ radioactive image used in training
- black-box — no access to the classifier but access to the output logits
  - loss of original vs radioactive

[Sablayrolles, ICML20]

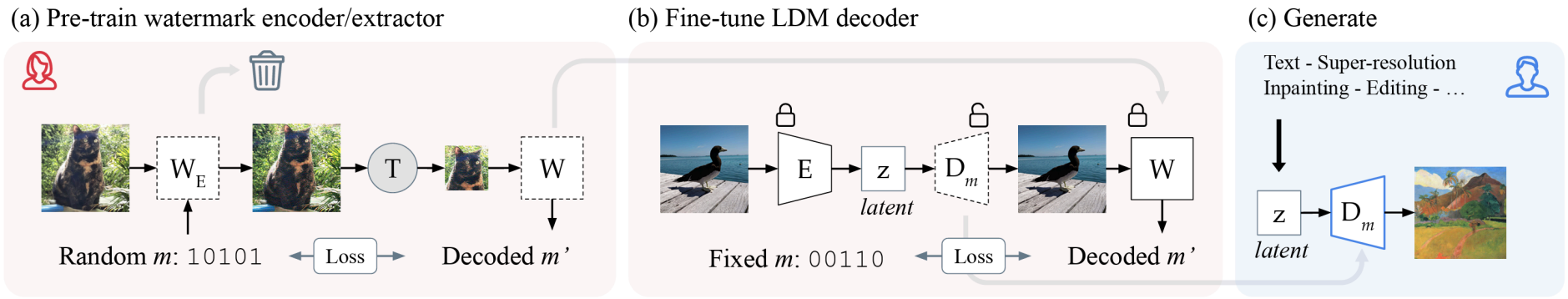
watermarking of models

- weight-based watermarks [Nagai, IJMIR18]
  - $L(f(x;w),y,w) = L_0(f(x;w),y) + \lambda L_R(w)$
  - does not compromise performance
  - robustness to fine-tuning

- query-based watermarks [Szyller, MM21]
  - provide modified responses (through some API)
    - hash function for selection of data points
    - hash function for output permutation
  - verify the examined model on selected data points
  - robustness to model stealing attacks, compromises performance

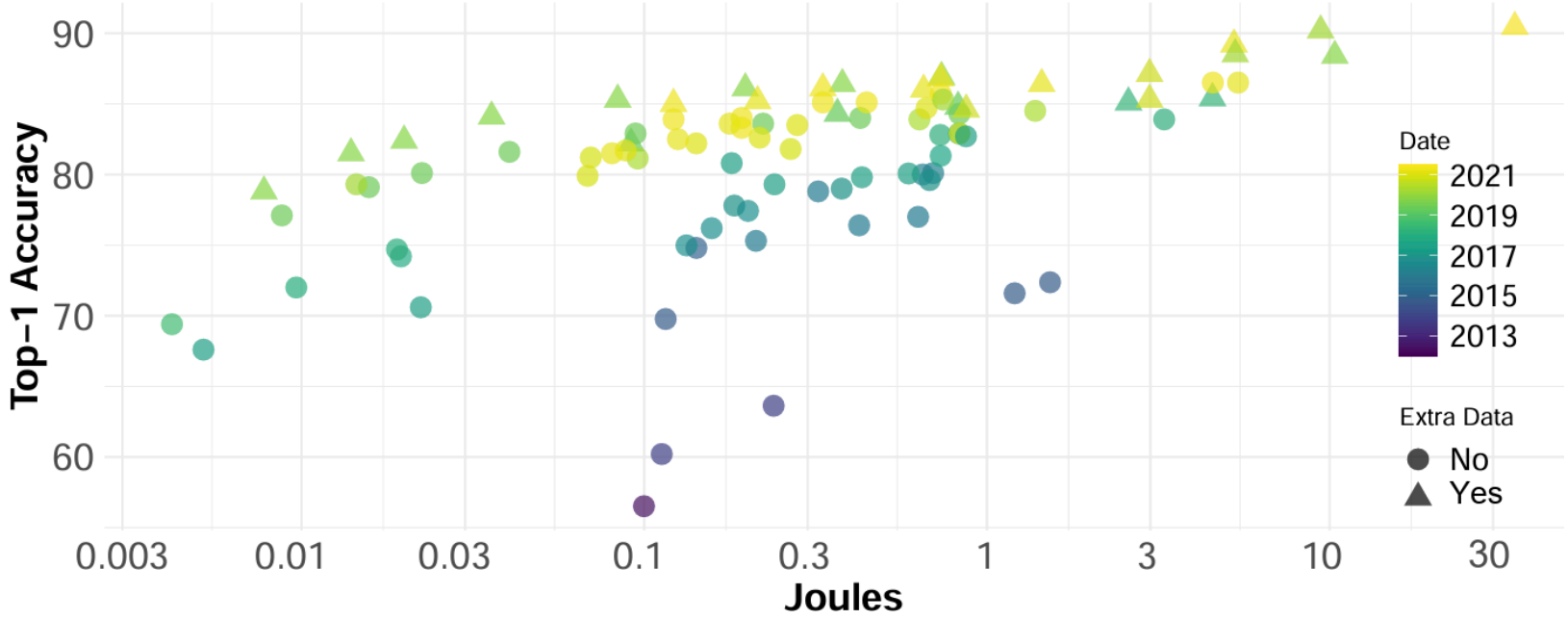
deep networks can overfit to incorrect labels

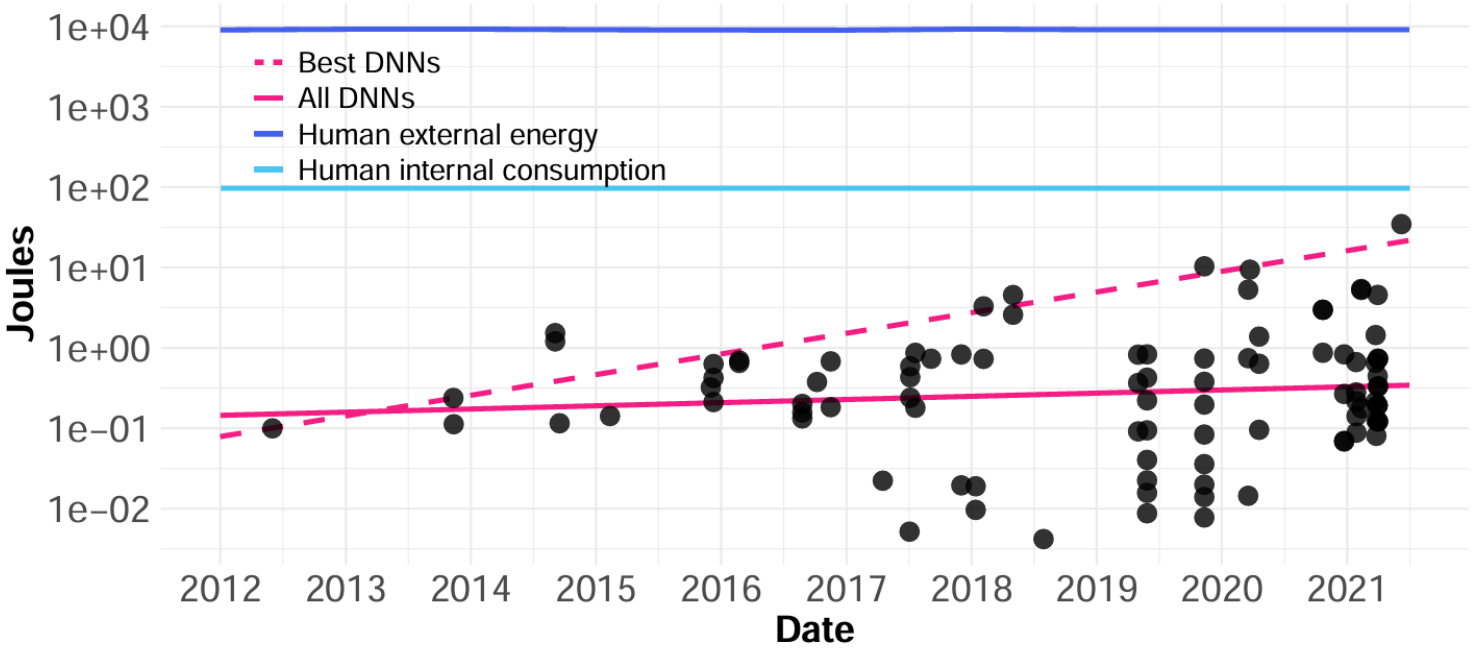
watermarking generated images



[Fernandez, ICCV23]

enviromental impact





[Desislavov, arxiv'23]

accountability

who is responsible when an AI system causes harm, and how can this responsibility be made traceable, enforceable, and useful?