Reading group presentation

Deep k-Nearest Neighbors

Towards Confident, Interpretable and Robust Machine Learning

Nicolas Papernot and Patrick McDaniel

Overview

- Motivation
    - *robustness*
    - *model confidence*
    - *interpretability*
- Proposed method
    - *DkNN*
- Evaluation

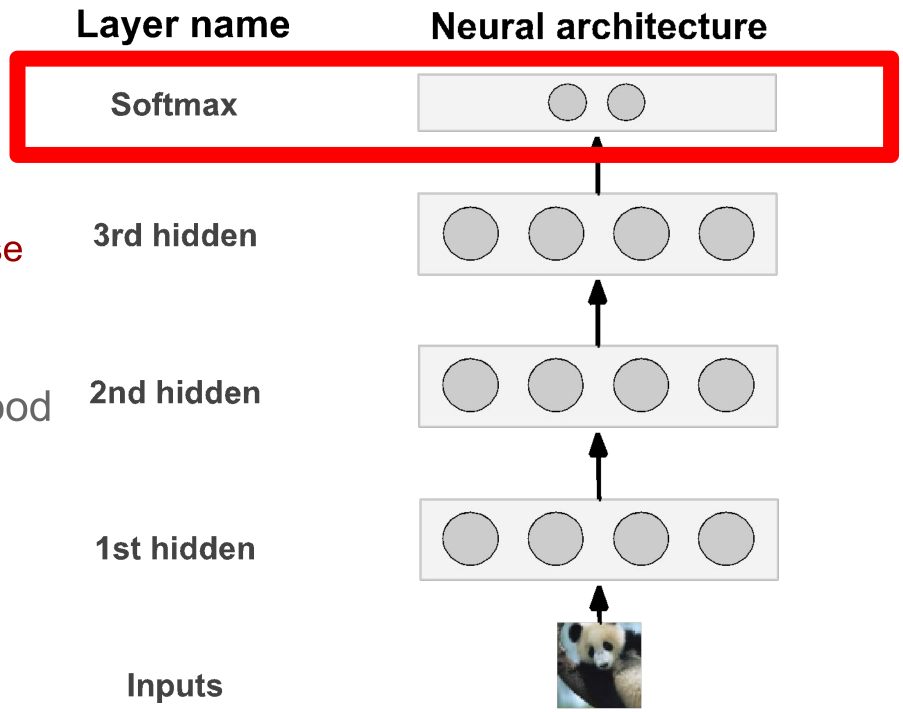
Motivation

What is the matter with vanilla softmax classification?

Vanilla Classification

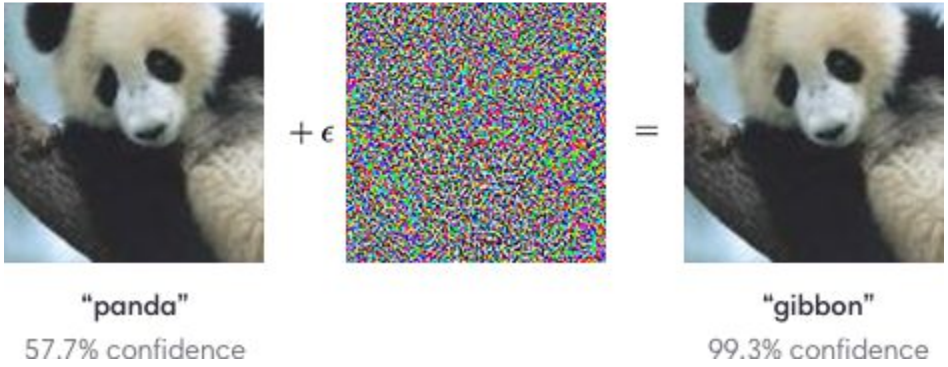
- softmax output = “confidence values”
- **Issues**:
    - Prediction isn’t robust
        - small perturbation to input can cause misclassification
        - dataset shift
    - Higher confidence → greater likelihood of correctly assigned label?
        - not necessarily true…
    - Interpretability: why this prediction?
        - black-box
        - GDPR - <u>need</u> interpretable output

\+ *proposed EU regulations on AI systems (new!)*



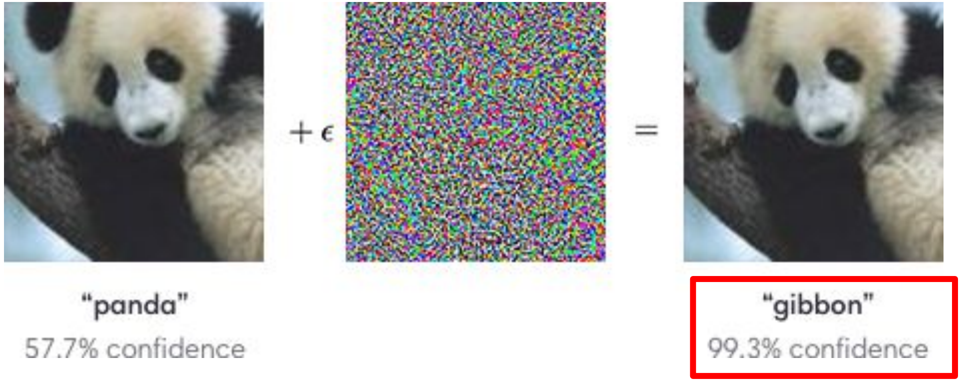
Robustness

- DNNs are vulnerable to input perturbations, which can cause them to misclassify
- **Adversarial examples**
  - perturb input images, without changing semantic meaning
    - *To the human eye, the adversarial example would still look like a panda.*
  - but causes model to misclassify...



Model confidence

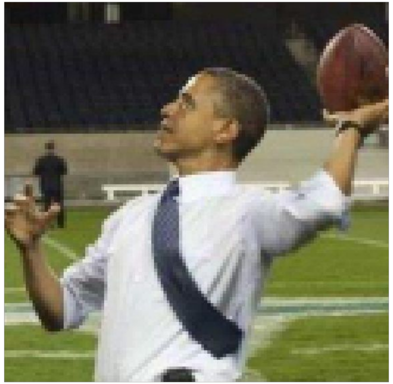
- Probabilities output by DNNs are **bad** indicators of model confidence



Model is **more** confident on adversarial example!

Interpretability

- Lack of explanations on model predictions
  - Often very important in evaluating **fairness in ML**!



Is this prediction based on a racial bias?

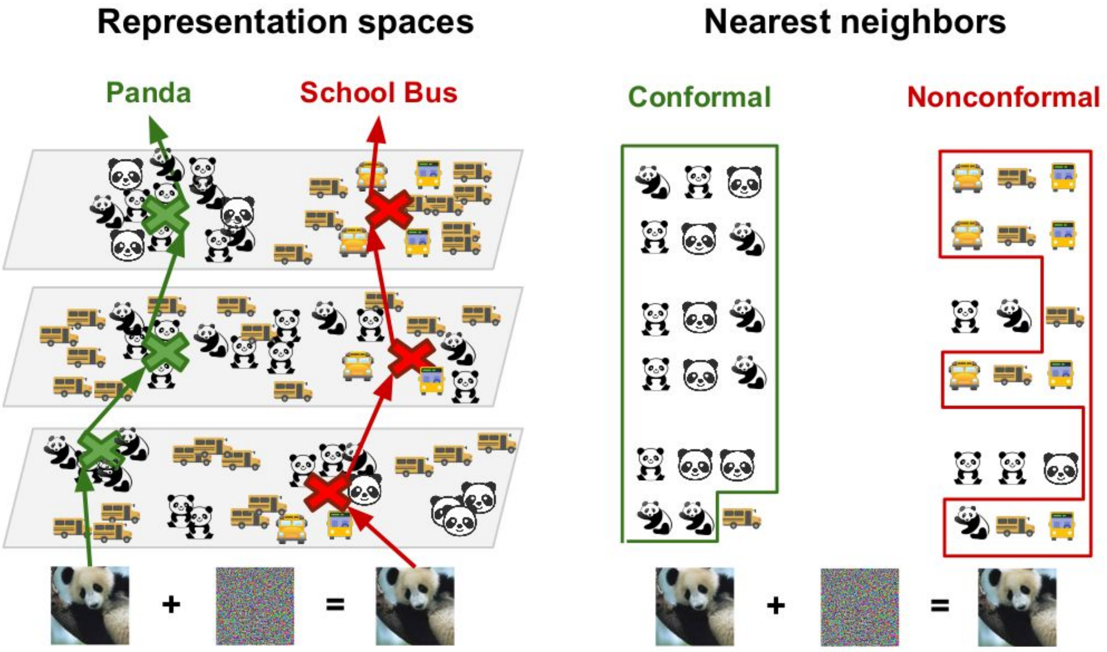
Prediction: Basketball (68%)

Proposed method

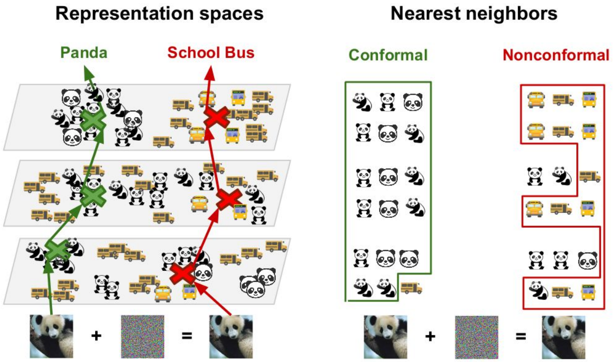
DkNN in a nutshell

- **new inference method**
- Exploits **hidden layer representations**
  - uses k-NN
  - analyzes nearest train sample labels

✓ more robust
✓ helps interpretability
✓ supports prediction w/ training samples



Deep k-Nearest Neighbors



x

k

f

training data

holdout data

DkNN

computes intermediate representations

+LSH

computes *non-conformity*

outputs *prediction*, confidence, *credibility*

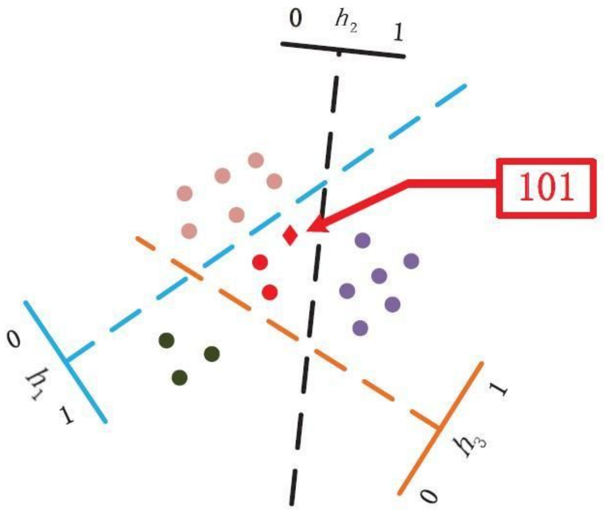
Likelihood of prediction being correct given the training set.

How relevant is the training set to the prediction?

Locality-Sensitive Hashing (LSH)

- Outputs of intermediate layers are often *high-dimensional*

- LSH is an efficient algorithm for looking up neighboring representations in high-dimensional spaces.
  - designed to *maximize* the collision of similar hashes
    - contrary to cryptographic hashes...
  - nearest neighbors according to cosine similarity



Non-conformity

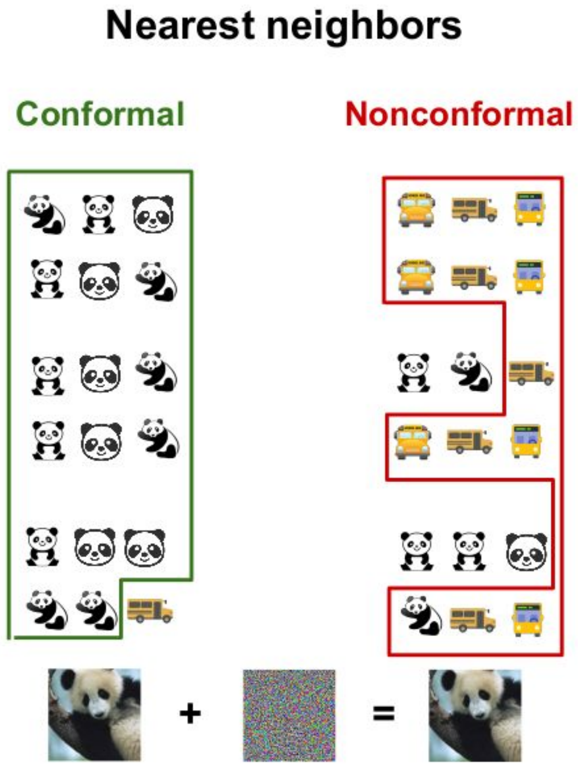
- Measures how ***different*** a ***test input*** is from ***training samples*** with the ***same label.***
- Non-conformity of input *x* with label *j*:

$$\alpha(x, j) = \sum_{\lambda \in 1..l} |i \in \Omega_\lambda : i \neq j|$$

… where Ω_λ is the multi-set of labels for the training points whose representations are closest to the test input's at layer λ.

What is the non-conformity of these pictures?

Where j = “panda”



Credibility

- Measures the ***support*** of ***predicted label j*** with regards to the ***training data***.
  - uses **non-conformity** to do this

- We want low credibility…
  - on *out-of-distribution data*
  - and *adversarial examples*

```
Input: x (test sample)

Output: credibility for all
possible labels j

1.  Separate a holdout dataset
    (X^H,Y^H) from the test set.
2.  Compute nonconformity values
    on (X^H,Y^H) → A.
3.  For each possible label j,
    compute the non-conformity of
    x and the credibility of
    label j:
```

$$\text{credibility}_j(x) = \frac{|\{\alpha \in A \colon \alpha \geq \alpha(x,j)\}|}{|A|}$$

Prediction

- The label with the **highest credibility**.

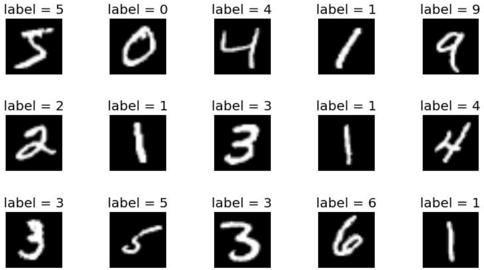
$$\arg\max_j \left(\text{credibility}_j(x)\right)$$

Dataset	DNN Accuracy	DkNN Accuracy
MNIST	99.2%	99.1%
SVHN	90.6%	90.9%
GTSRB	93.4%	93.6%

Limited or no impact on accuracy!

Evaluation

Datasets



- MNIST
  - digits (0-9)

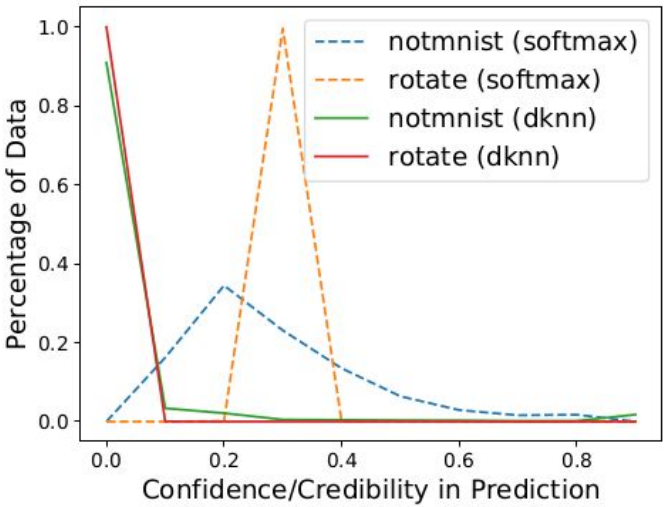
- SVHN
  - colored house numbers (0-9)

- GTSRB
  - traffic sign images

Credibility

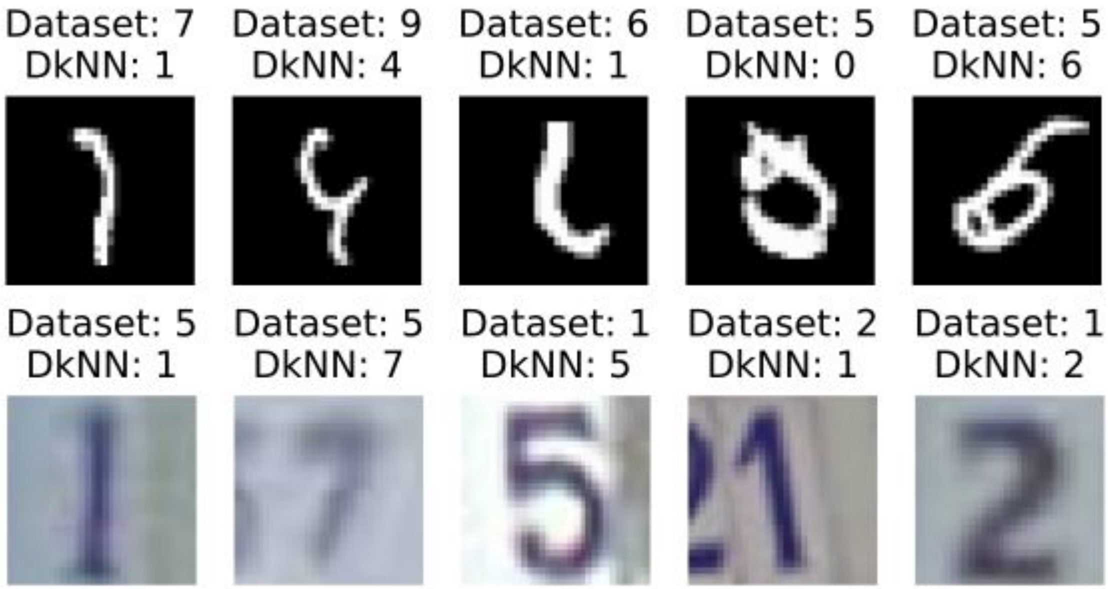
✓ Is lower on out-of-distribution samples.

- in-distribution: MNIST
- out-of-distribution: NotMNIST (unicode chars) + rotated MNIST



Credibility (2)

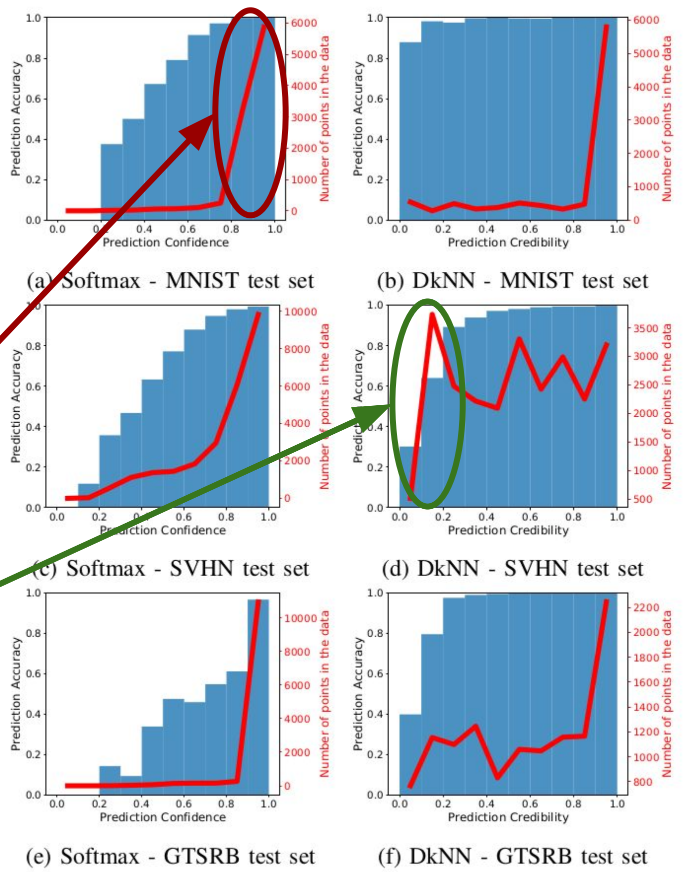
✓ Can even find test inputs whose label in the *original dataset* is ***wrong***.



Credibility (3)

- well-calibrated credibility should be linearly aligned with accuracy
    - high credibility → greater probability of correct prediction → high accuracy
- softmax seems to have this linear property
    - too confident on majority of test data…
- if the task is difficult, DkNN assigns lower credibility

Softmax DkNN



(a) Softmax - MNIST test set
(b) DkNN - MNIST test set
(c) Softmax - SVHN test set
(d) DkNN - SVHN test set
(e) Softmax - GTSRB test set
(f) DkNN - GTSRB test set

Interpretability

✓ DkNNs give ***explanations by example***

- ○ *examples* are training samples whose representations are *closest* to the test sample

Not only skin color, but ball might also contribute to misclassification!

Prediction: Basketball (68%)

Robustness

Dataset	DNN Accuracy	DkNN Accuracy
MNIST	99.2%	99.1%
SVHN	90.6%	90.9%
GTSRB	93.4%	93.6%

✓ Less adversarial examples are misclassified

- adversarial example classification accuracy is **higher**

Dataset	Attack	Attack Parameters	DNN	DkNN
MNIST	FGSM	ε=0.25	27.1%	54.9%
	BIM	ε=0.25, α=0.01, i=100	0.7%	16.8%
	CW	κ=0, c=10^{-4}, i=2000	0.7%	94.4%
SVHN	FGSM	$\varepsilon = 0.05$	9.3%	28.6%
	BIM	ε=0.05, α=0.005, i=20	4.7%	17.9%
	CW	κ=0, c=10^{-4}, i=2000	4.7%	80.5%
GTSRB	FGSM	$\varepsilon = 0.1$	12.3%	22.3%
	BIM	ε=0.1, α=0.005, i=20	6.5%	13.6%
	CW	κ=0, c=10^{-4}, i=2000	3.0%	74.5%

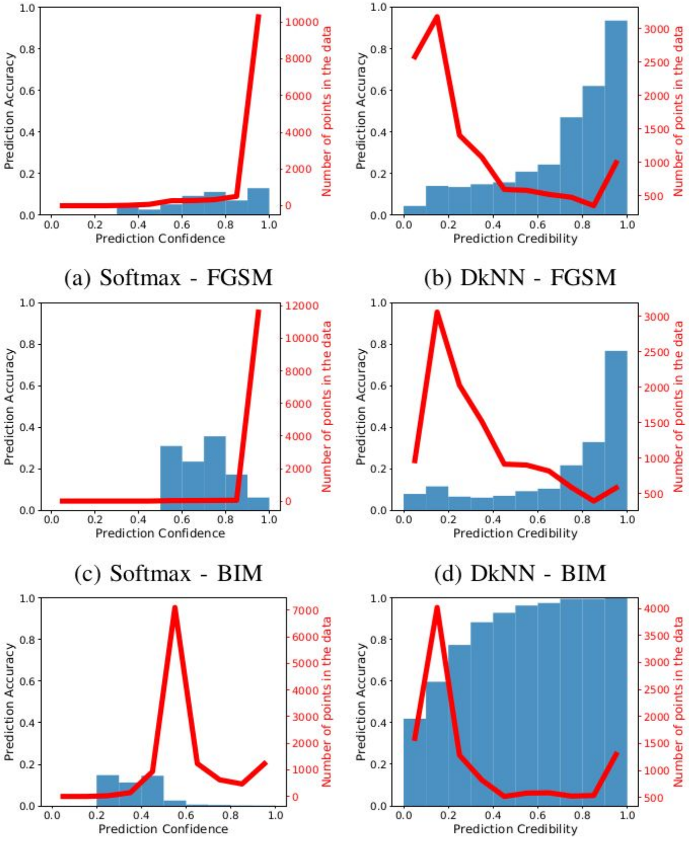
For most adversarial examples credibility is < 0.5.

Robustness (2)

- Softmax
  - high confidence on majority of adversarial examples
- DkNN
  - credibility is < 0.5 for most adversarial examples

Softmax

DkNN



(a) Softmax - FGSM

(b) DkNN - FGSM

(c) Softmax - BIM

(d) DkNN - BIM

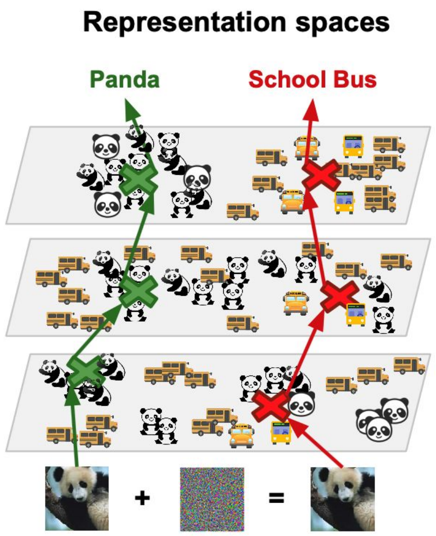
(e) Softmax - CW

(f) DkNN - CW

Robustness (3)

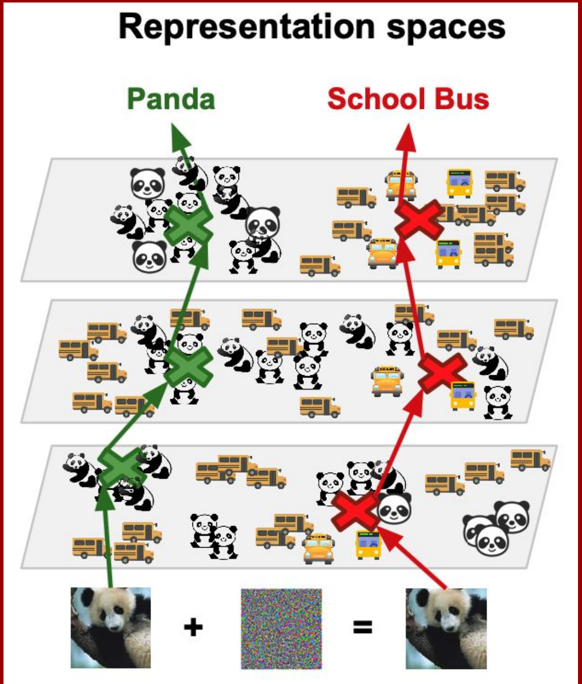
✓ *Adaptive adversarial attacks* against DkNNs are not imperceptible.

- **Feature adversaries:** Forces hidden layer representation of adversarial example to resemble training data representations of the desired label.
    - e.g. panda in hidden layers should look more like a school bus
        - adversarial panda morphs into a school bus, making the attack obvious



This happens despite small perturbation budget!

Conclusions



- DkNN exploits intermediate representations to make predictions
  - withstand adversarial perturbation better (*robustness*)
  - that are supported by training data (*credibility*)
  - *interpretable* by providing similar examples
- Questions that remain open:
  - Metric that measures likelihood of prediction being correct, given the training set.
    - ~~confidence~~