A Duty to Forget, a Right to be Assured?
Exposing Vulnerabilities in Machine Unlearning Services

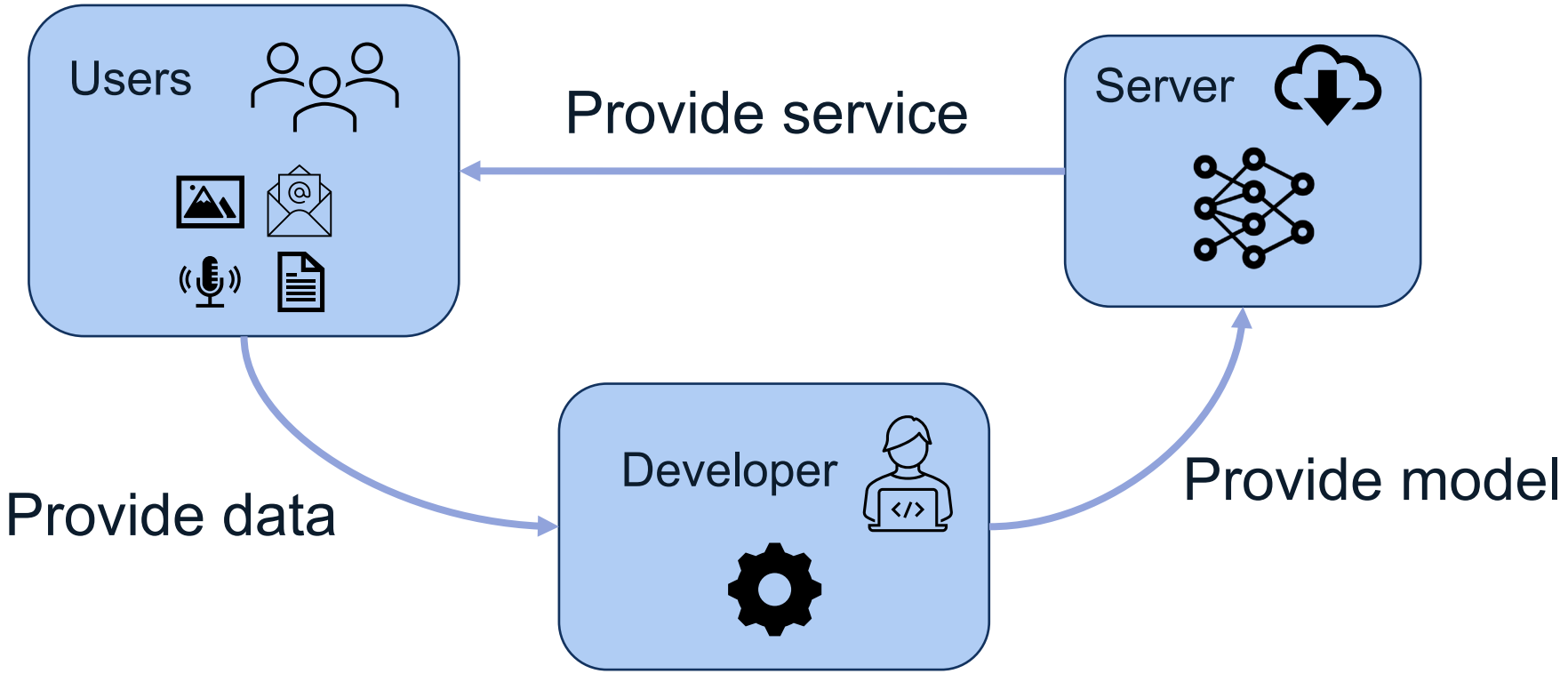
Hongsheng Hu* (CSIRO), Shuo Wang (CSIRO), Jiamin Chang (University of New South Wales), Haonan Zhong (University of New South Wales), Ruoxi Sun (CSIRO), Shuang Hao (University of Texas at Dallas), Haojin Zhu (Shanghai Jiao Tong University), and Minhui Xue (CSIRO)

NDSS SYMPOSIUM/2024

Presented by
Internet Society

#NDSSSymposium2024

Machine learning as a service



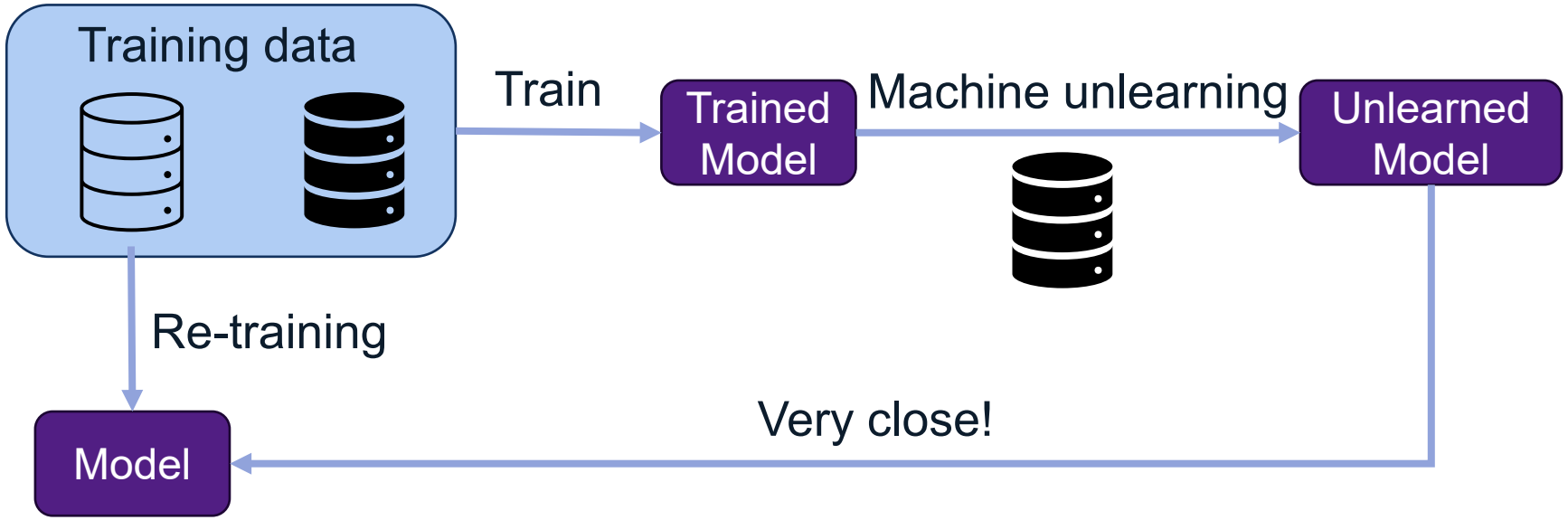
What if users “regret”?

“I want my data to be deleted”

- Data privacy
- Data security
- Control over personal information
- Past mistakes or embarrassing information
- Legal requirements, e.g., GDPR

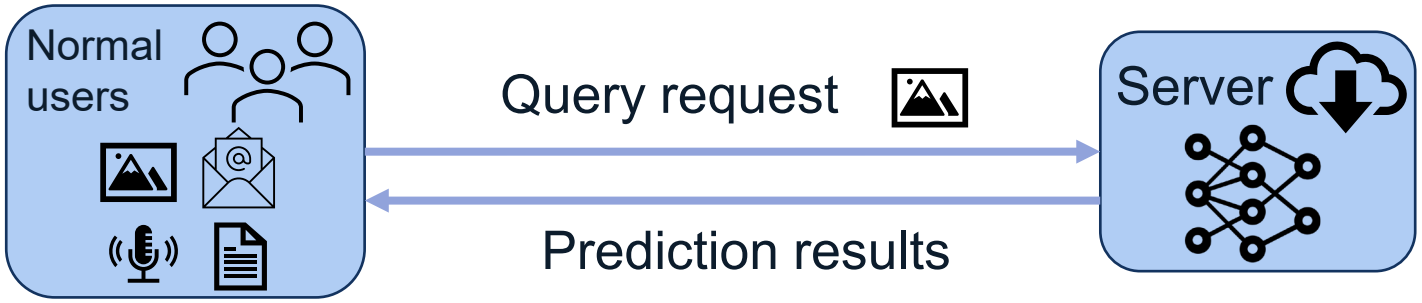
Machine unlearning

Removing the inference of unlearned data efficiently and effectively

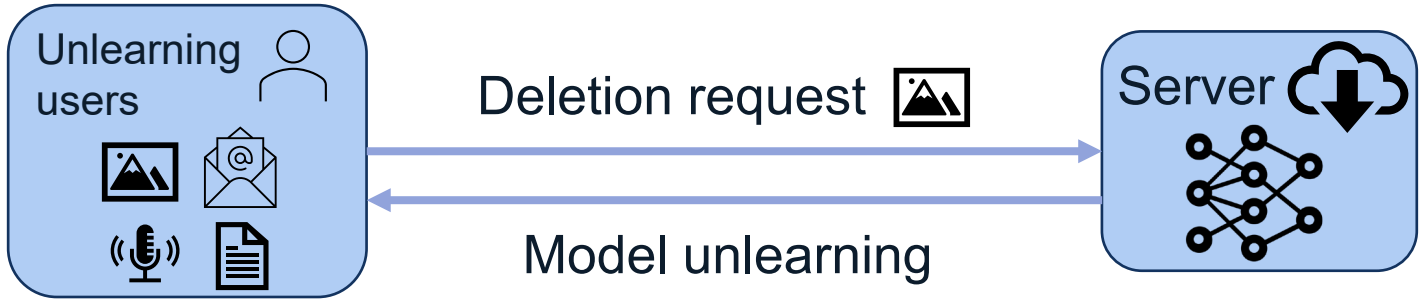


Machine unlearning services

Normal business

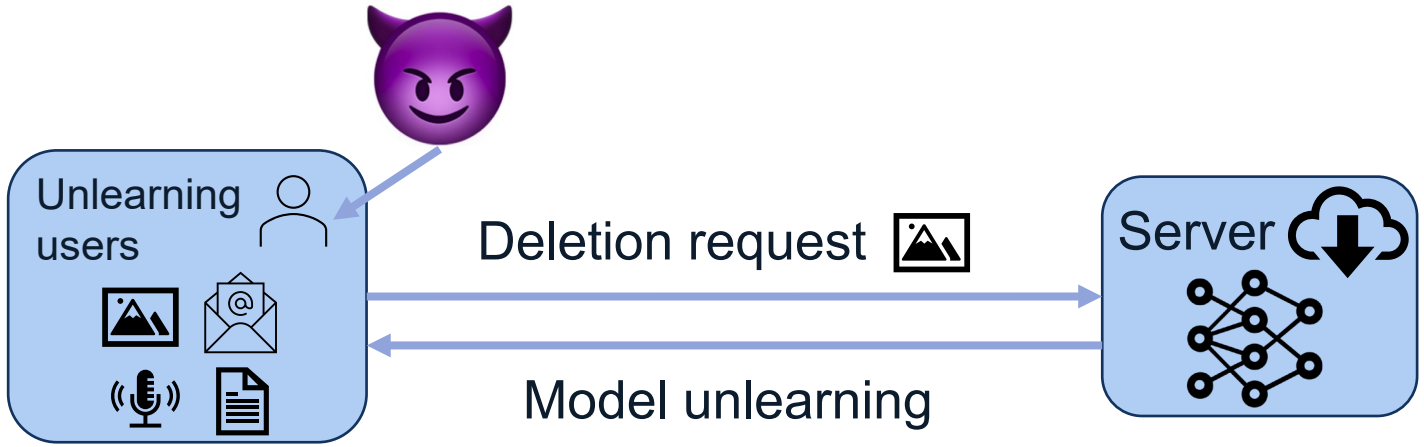


Machine unlearning services



Where could problems occur?

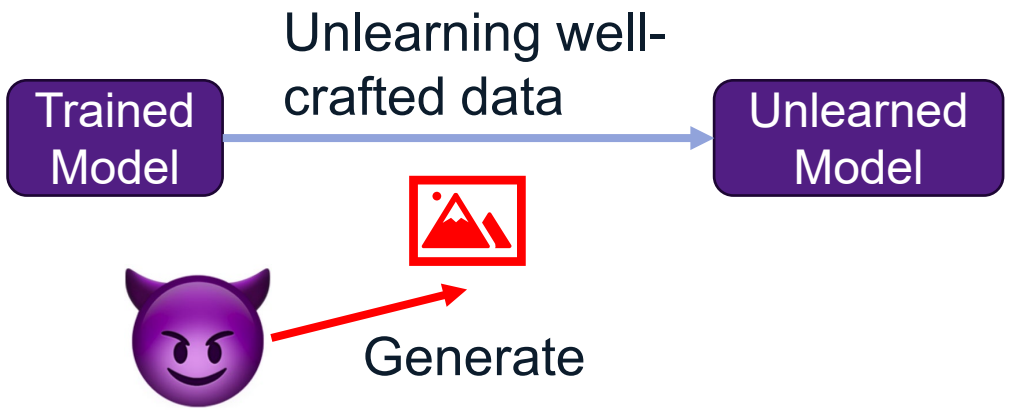
- What can be wrong if there is an unlearning user submitting potential malicious deletion requests?
- What consequence malicious deletion requests can bring?



Normal unlearning vs. over-unlearning

Normal unlearning
- Data is deleted
- Model utility is preserved or slightly reduced

Over-unlearning threat
- More information is deleted than expected!
- Model utility suddenly deteriorates



Technique challenges: how to achieve over-unlearning?

- Difficult to quantify how much information is contained in a data point
- Difficult to quantify how a data point may influence the model

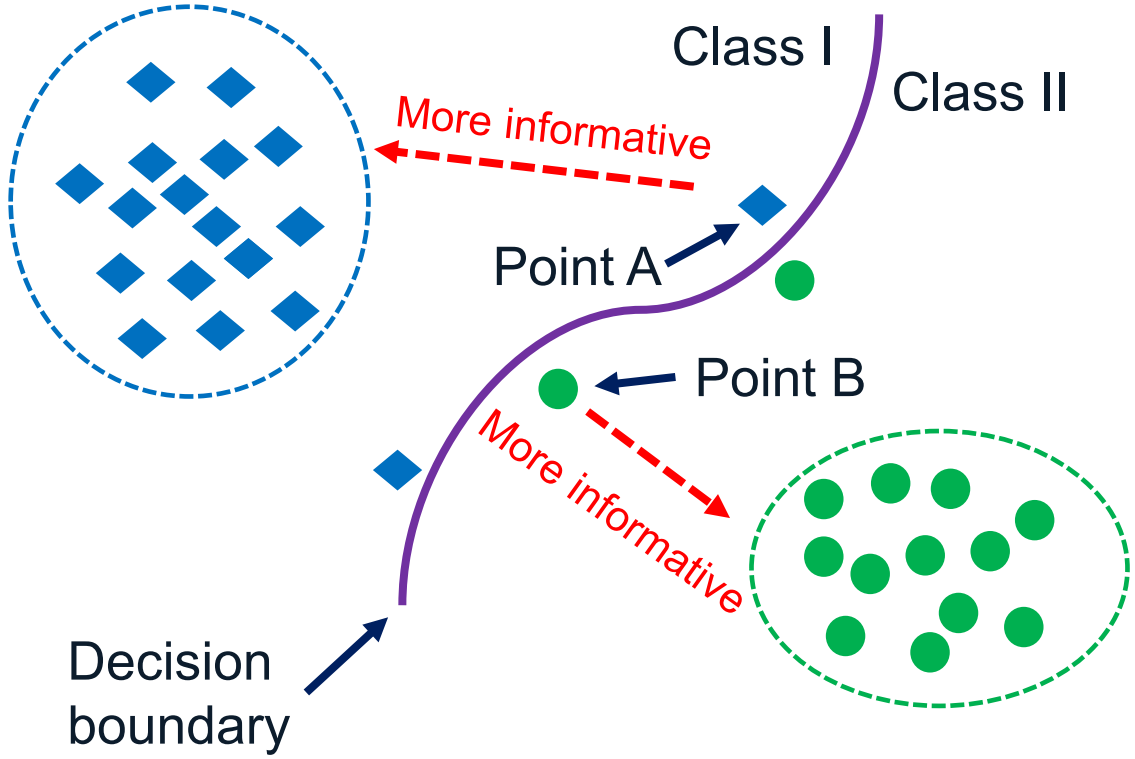
Blending as naïve over-unlearning

- Intuition: Incorporate additional sample (x_b) information into the unlearned sample (x) via blending
- $x' = a \cdot x + (1 - a) \cdot x_b$; a is a blending parameter

Unlearned sample Additional sample Blended sample

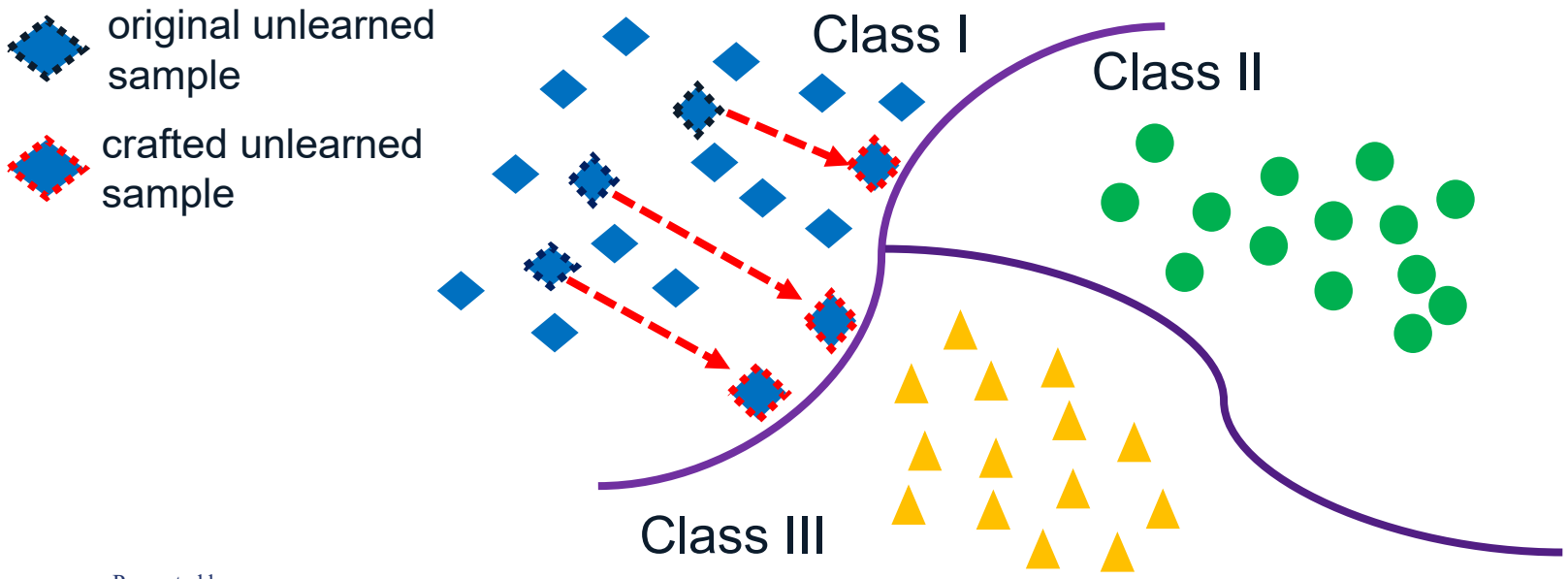
Pushing as advanced over-unlearning

Key intuition: Points near the decision boundary is more informative than the points far away from that



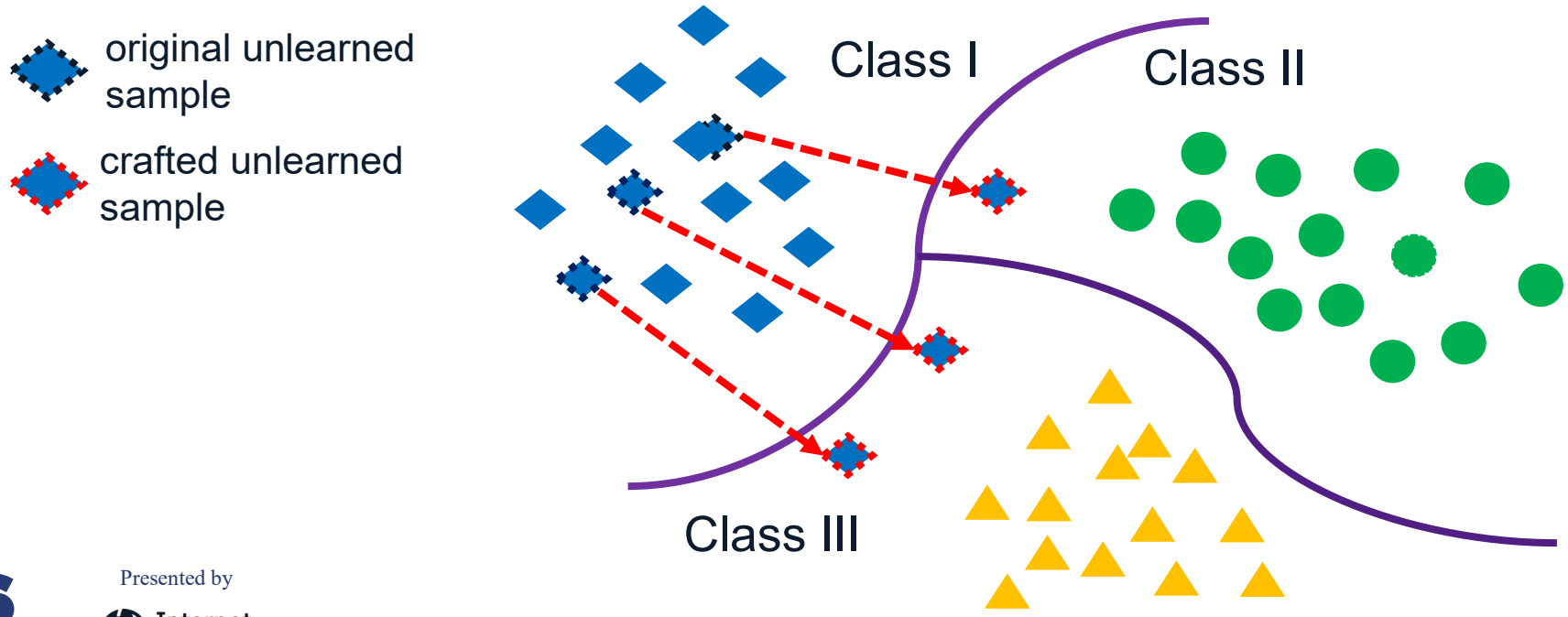
Two pushing strategies

- **Pushing-I**: push the unlearned sample near the decision boundary but not across it.



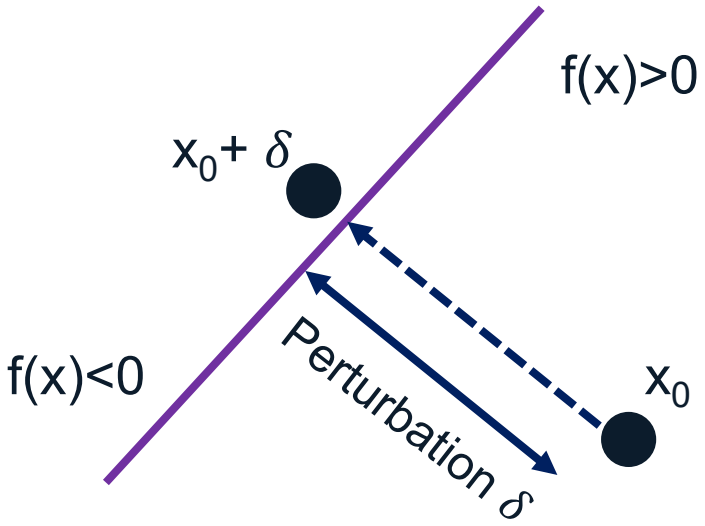
Two pushing strategies

- **Pushing-II**: push the unlearned sample just across the decision boundary.



How to achieve Pushing?

- Moving a sample towards the decision boundary is well studied in adversarial example attacks.
- We use *black-box adversary attack techniques* to achieve pushing



Pushing implementation

- $x' = x + \delta$; add small perturbation to the original unlearned sample
- $dis(x', \theta) < \varepsilon$; ensure the crafted sample is close enough to the decision boundary of the model θ

Experimental setup

- Utility metric: accuracy of the model
- Baseline: normal unlearning
- Unlearning method: approximate unlearning methods
- Number of unlearned sample: no more than 50% of a class
- Blending: how blending “B” to “A” influences the model’s accuracy on “B”
- Pushing: how moving “A” to the decision boundary influences the model’s accuracy on “A”

Is Blending effective?

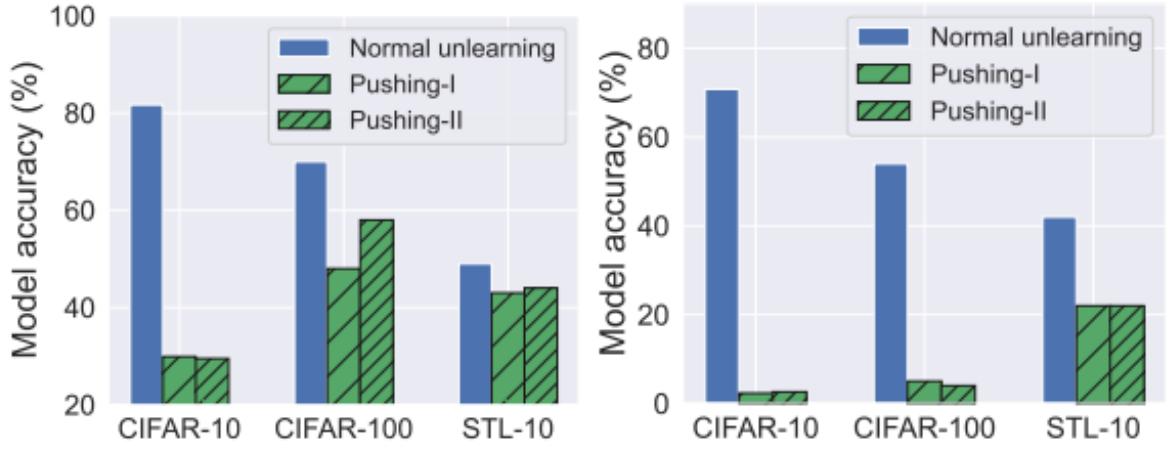
- Blending is not stable for over-unlearning
- Hypothesis: blending does not change the decision boundary of the model too much through unlearning

  - **Experimental setting**: unlearn samples of one class;
  - **% of unlearned samples**: percentage of the unlearned samples on the class;
  - **Acc_N**: accuracy of the normal unlearned model on the class;
  - **Acc_O**: accuracy of the over-unlearned model on the class.

Dataset	% of unlearned samples	Blending ratio	Acc_N - Acc_O
CIFAR-10	10%	0.5	1.4%
CIFAR-100	10%	0.5	<0
STL-10	10%	0.5	<0

Is Pushing effective?

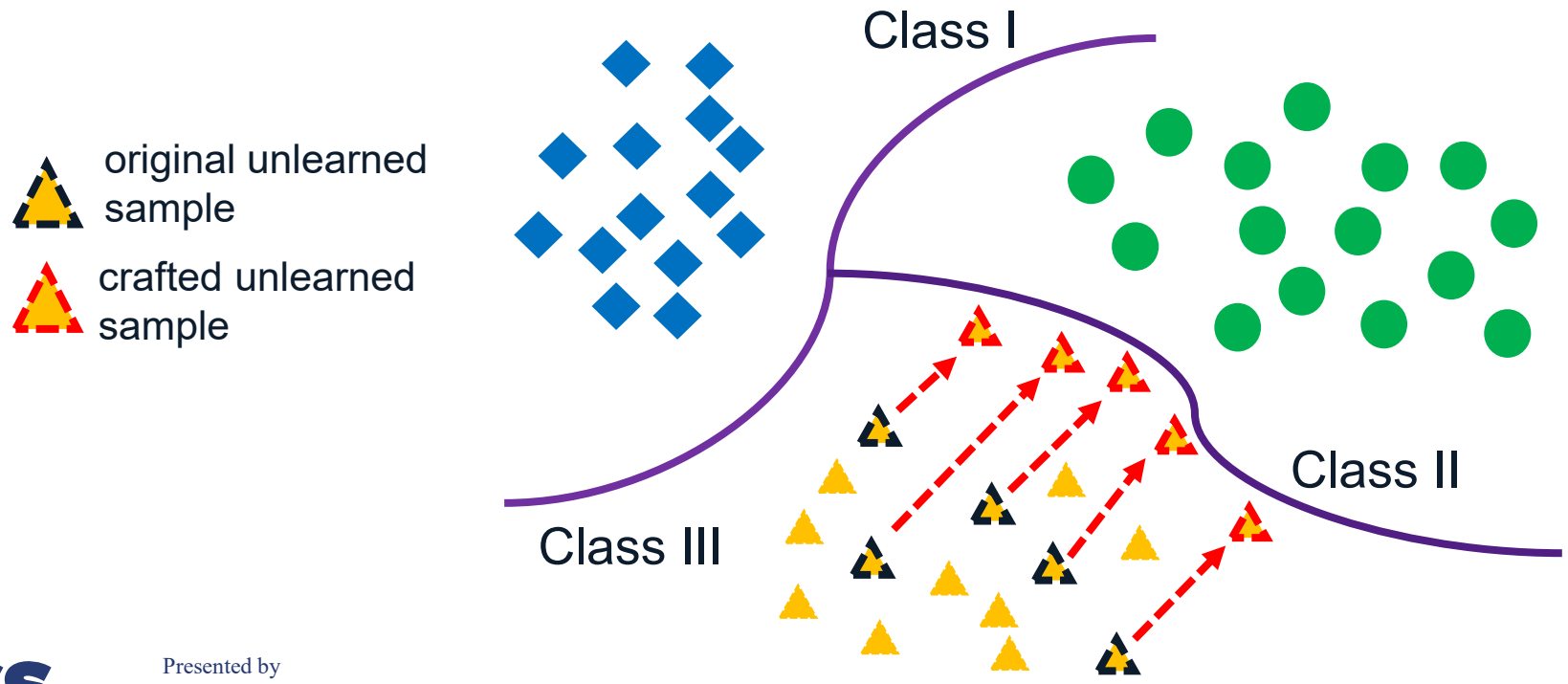
- Pushing is always effective!
- Data points close to decision boundary are important in unlearning!



(a) Unlearn 10% data of a class (b) Unlearn 50% data of a class

Pushing can be more dangerous than just utility reduction!

- What if pushing the unlearned samples to a particular decision region?



The “controlled” misclassification

- Class A: the label of the unlearned data
- Class B: the “target” label, i.e., moving all the unlearned samples to near the decision boundary of class B
- Pushing can make the unlearned model misclassify samples of A (1,000 in total) to B!

Status	Class A	Class B
Before unlearning	878	0
Normal unlearning	708	5 (A->B)
Pushing-I	26	378 (A->B)
Pushing-II	31	247 (A->B)

Takeaways

- Pushing is a reliable and effective way for over-unlearning. Data points near the decision boundary have high impact on machine unlearning.
- A larger number of unlearned samples enable more effective over-unlearning.
- Model's behaviour might be "controlled" through over-unlearning.

Discussion

- Possible defence
  - Hashing as a possible defence
  - Membership inference
  - Anomaly detection
- Can over-unlearning be success in exact unlearning?
  - Possible! Maybe through poisoning.
- More than model utility!
  - How malicious unlearning may affect model robustness?

Discussion

- Possible defence
  - Hashing as a possible defence
  - Membership inference
  - Anomaly detection
- Can over-unlearning be success in exact unlearning?
  - Possible! Maybe through poisoning.
- More than model utility!
  - How malicious unlearning may affect model robustness?

Thank you for your attention!
Questions?
Hongsheng. Hu@csiro.au