black hat USA 2021

August 4-5, 2021

BRIEFINGS

# Siamese Neural Networks for Detecting Brand Impersonation


Justin Grana (Presenter)

Yuchao Dai

Jugal Parikh

Nitin Kumar Goel

# Everyone knows the brand impersonation story


*Source: Bing Images*

# Human process for identifying brand impersonation

Task 1: Identify the spoofed brand (easy)

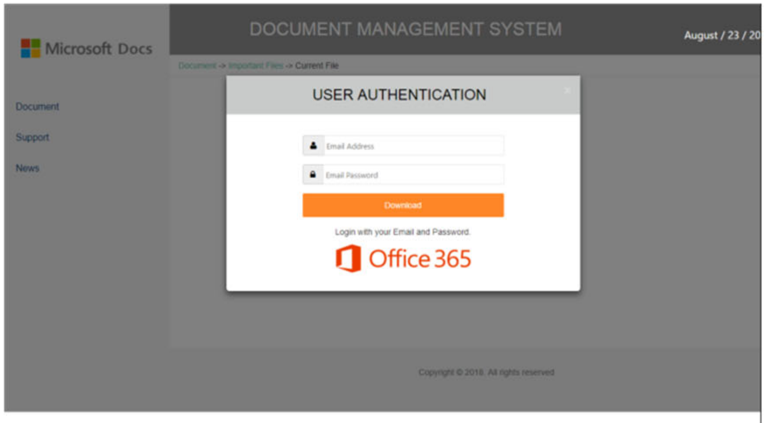
Task 2: Check “other” details to see if it aligns with the brand

- Domain names
- URLs
- Tone of the message…etc

- An automated filter would need to do both.
- This project focuses on training a machine learning model to perform task 1, a pre-requisite for task 2.

# Data

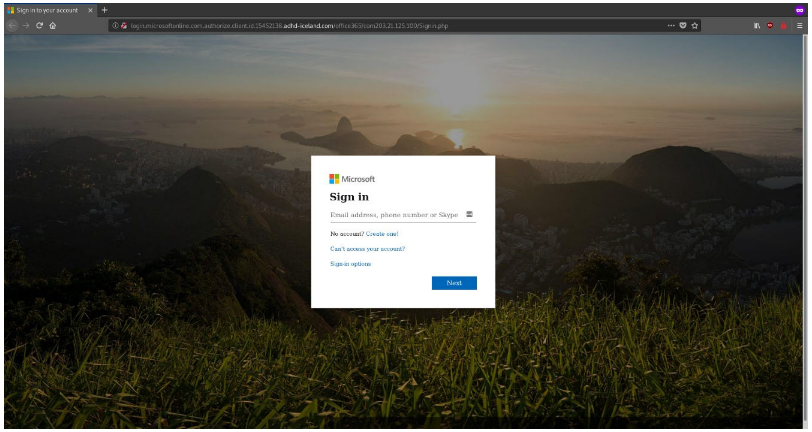
- Detonation service screenshots of known malicious brand impersonations.




- 50K + images with over 1.3K unique brands
- How can we succeed in classification without non-malicious content?

# Underlying Assumption

- The best brand impersonation content will look identical to the true brand content.



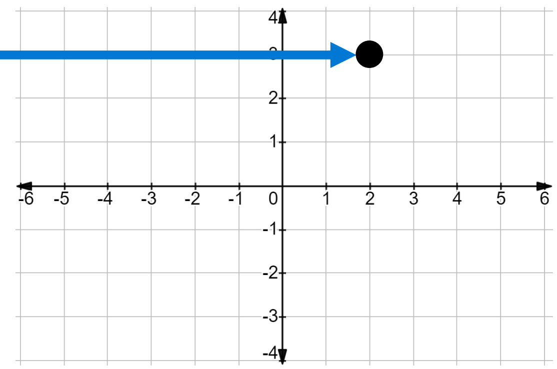

- The *best* we can hope to do using visual attributes *alone* is identify brands, not conduct a benign/malicious classification.
- This is fundamentally a **multi-class classification problem**

# Possible approaches

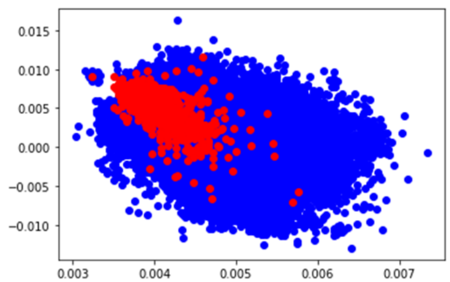
- Image Hashing
  - Too many variations on the same brand
- Traditional classification (i.e. feed forward neural networks)
  - Too many classes with too few observations per class
- “Few-shot” learning
  - Siamese Networks

# What is an Embedding?

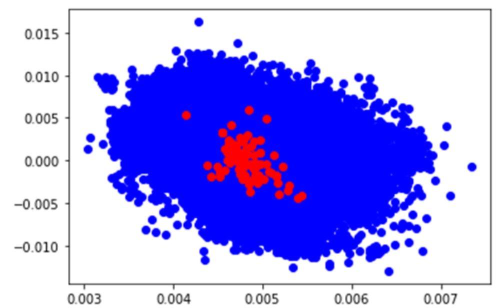
Microsoft




Paypal

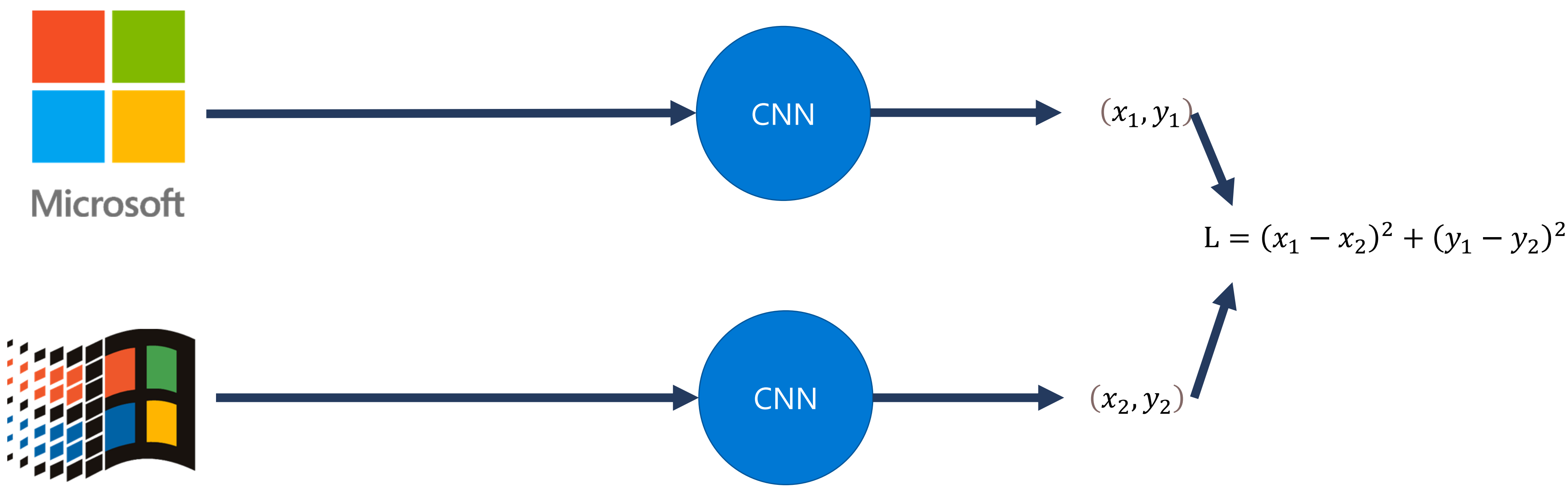



TD



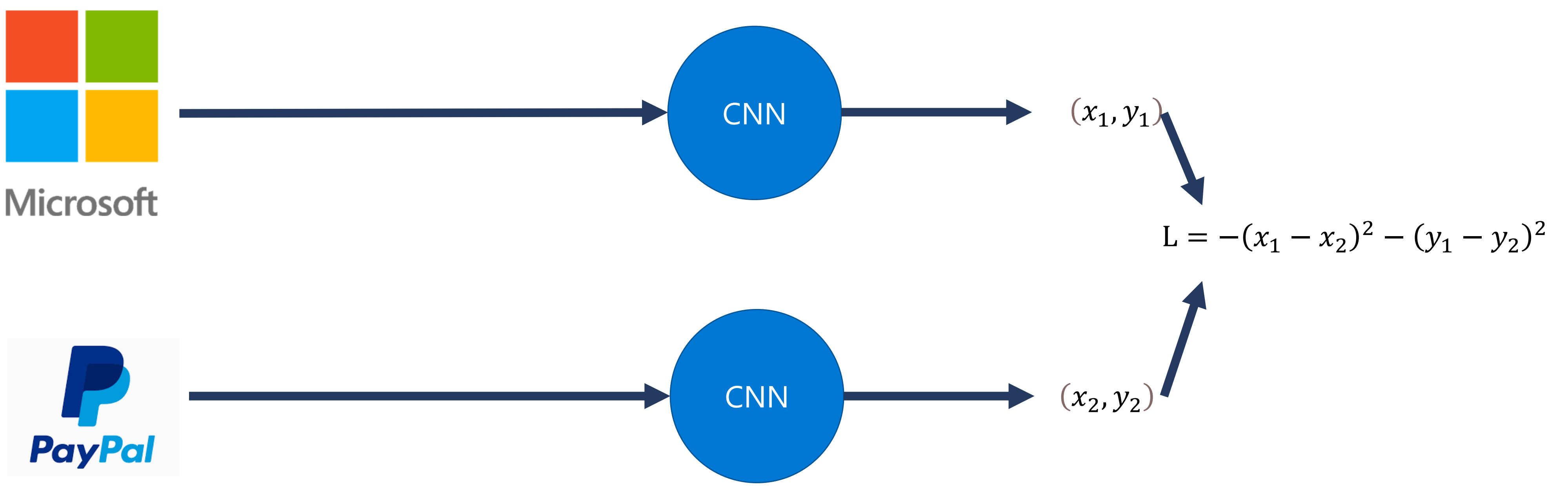
# Siamese Networks with Contrastive Loss

- For inputs of the same brand:

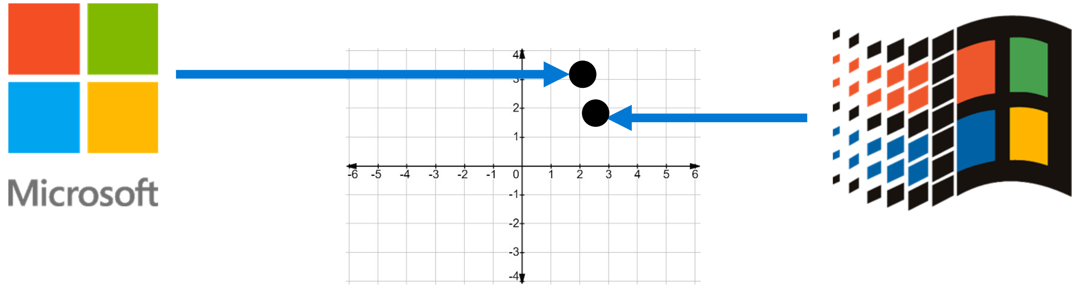


# Siamese Networks with Contrastive Loss

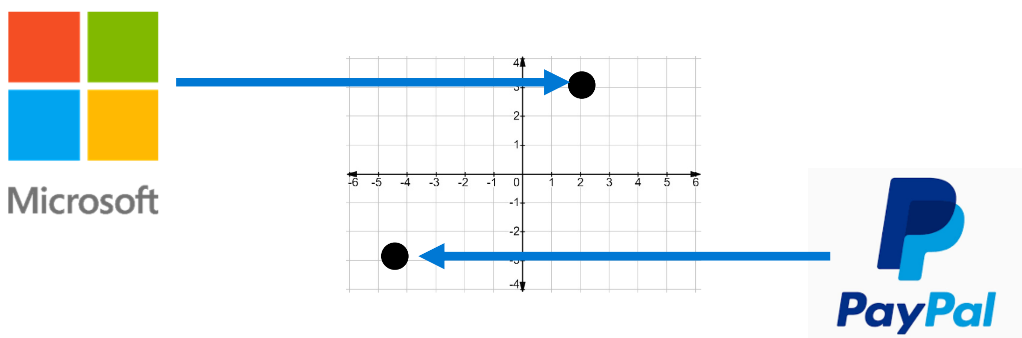
- For inputs of the different brands:



# Result




Good: Images from the same brand close together




Good: Images from different brands far apart

# Outcome-Motivated Metrics

| | True | | Predicted |
|---|---|---|---|
| Know Hit Rate | Microsoft | → | Microsoft |
| Known Miss Rate | Microsoft | → | PayPal |
| Incorrect Unknown Rate | Microsoft | → | ? |

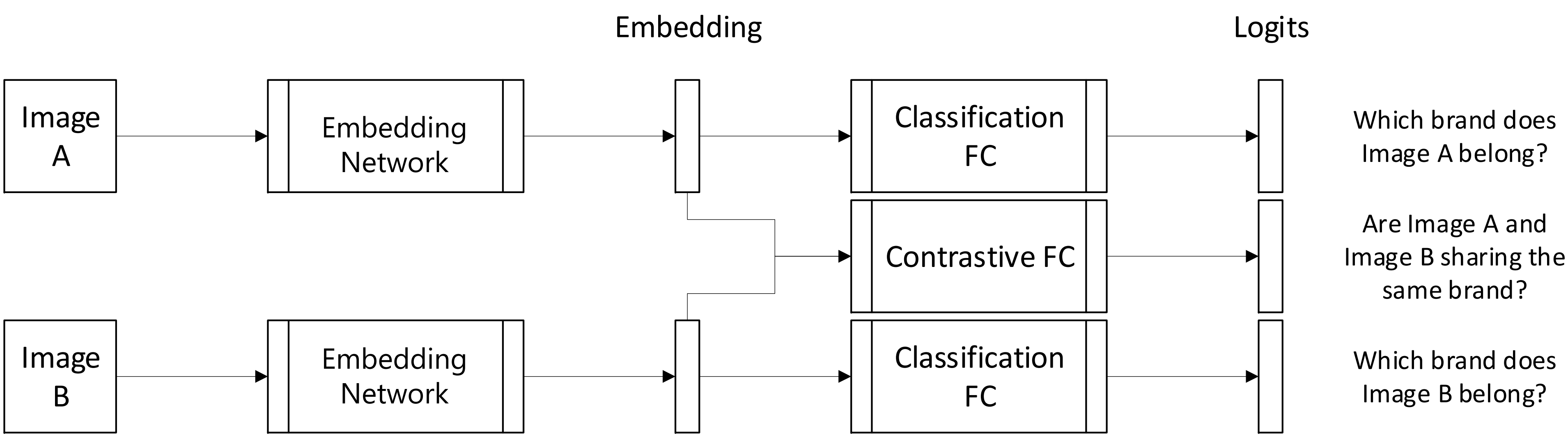
| | True | | Predicted |
|---|---|---|---|
| Unknown Misclassification | ? | → | Microsoft |
| Correct Unknown Rate | ? | → | ? |

# Architecture

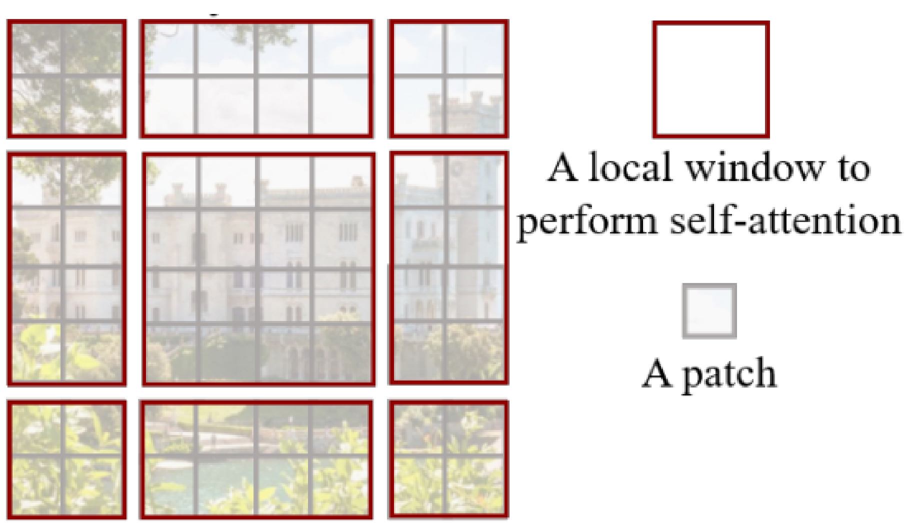
Parameters between two Pretrained Model are shared.

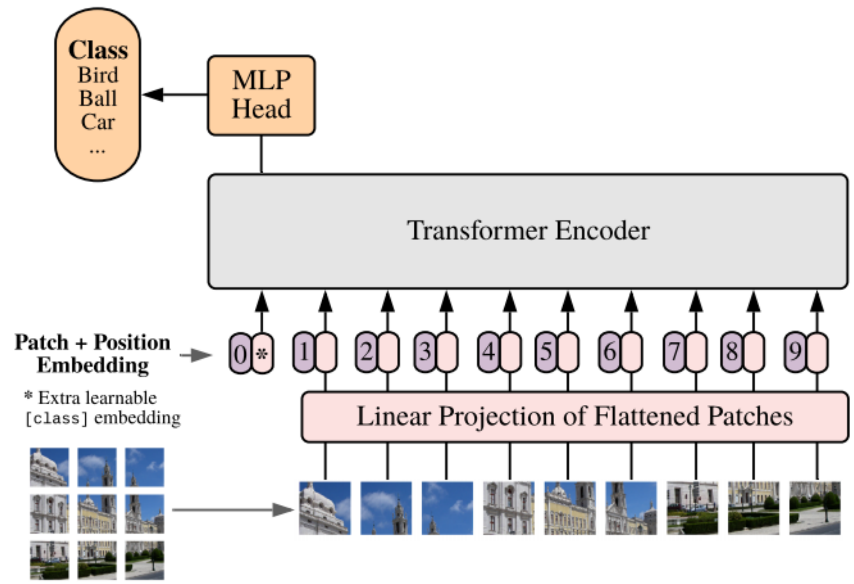
Parameters between two Classification FC are shared.

Final loss is the weighted sum of the 3 sub-losses.



# Swin Transformers



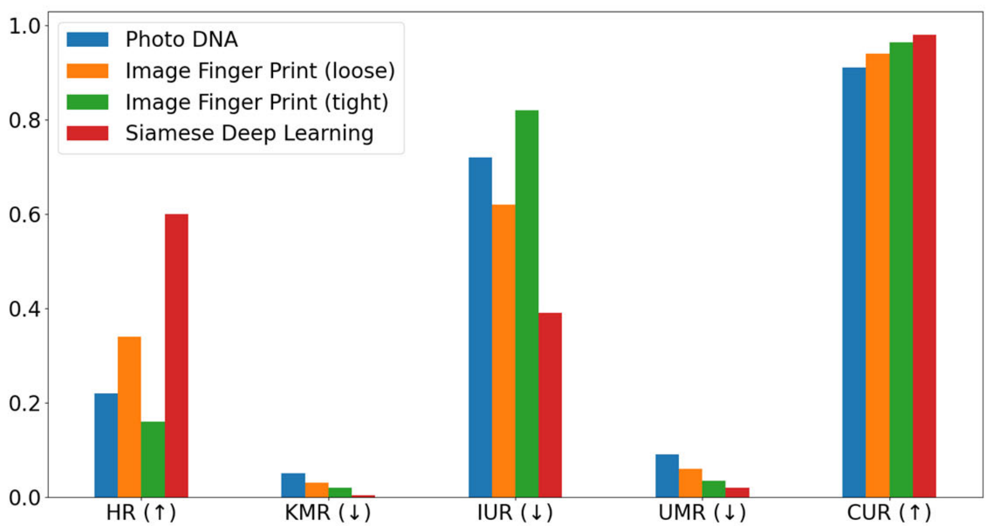

1. Split image into 4px*4px patches
2. Observe patches with shifted windows

1. Embed patches with linear projection
2. Feed patches to transformer

# Training Parameters

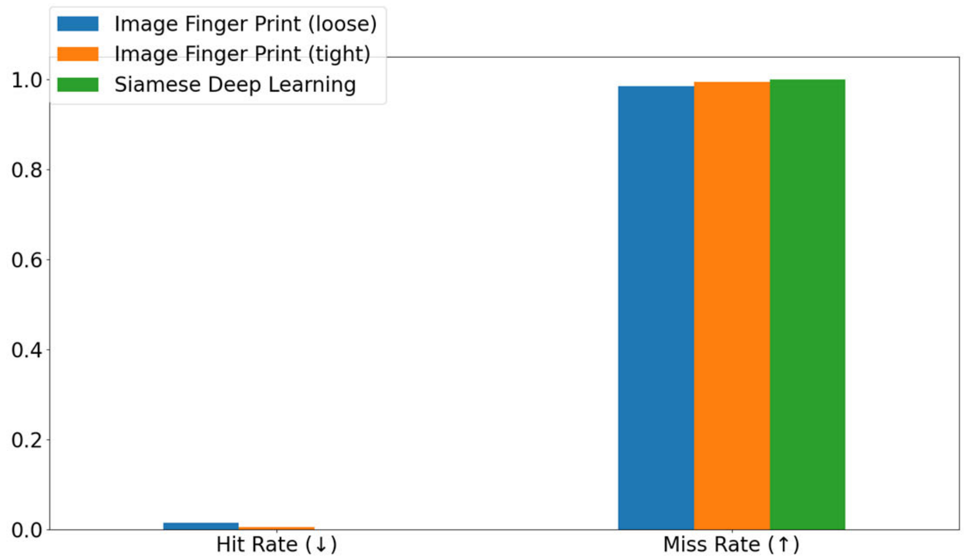
- 80/20 Split
  - >50k images with >1.3k unique brands
  - ~500 brands with only one screenshot, all in test set
- Three separate evaluations:
  - Test Set
  - Alexa Top 33k (hit/miss only)
  - Known bad (hit/miss only)

# Results on Held Out Set



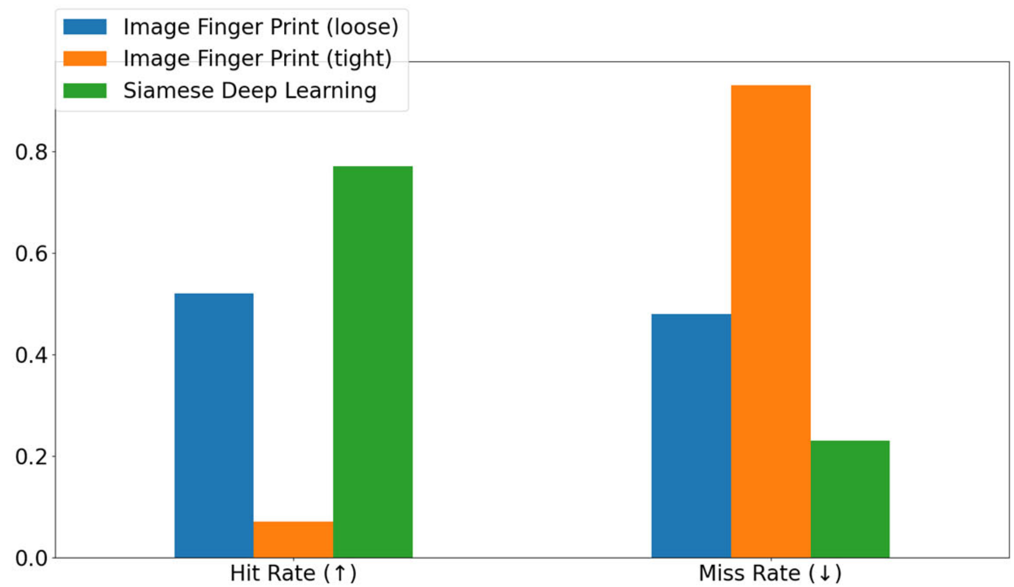

- HR
- KMR
- IUR ?
- UMR ?
- CUR ? ?

# Hits and Miss Rate ~30k most trafficked websites




These are known benign *home pages* so we would expect a good algorithm to *not* detect these as a brand impersonated *log-in page*

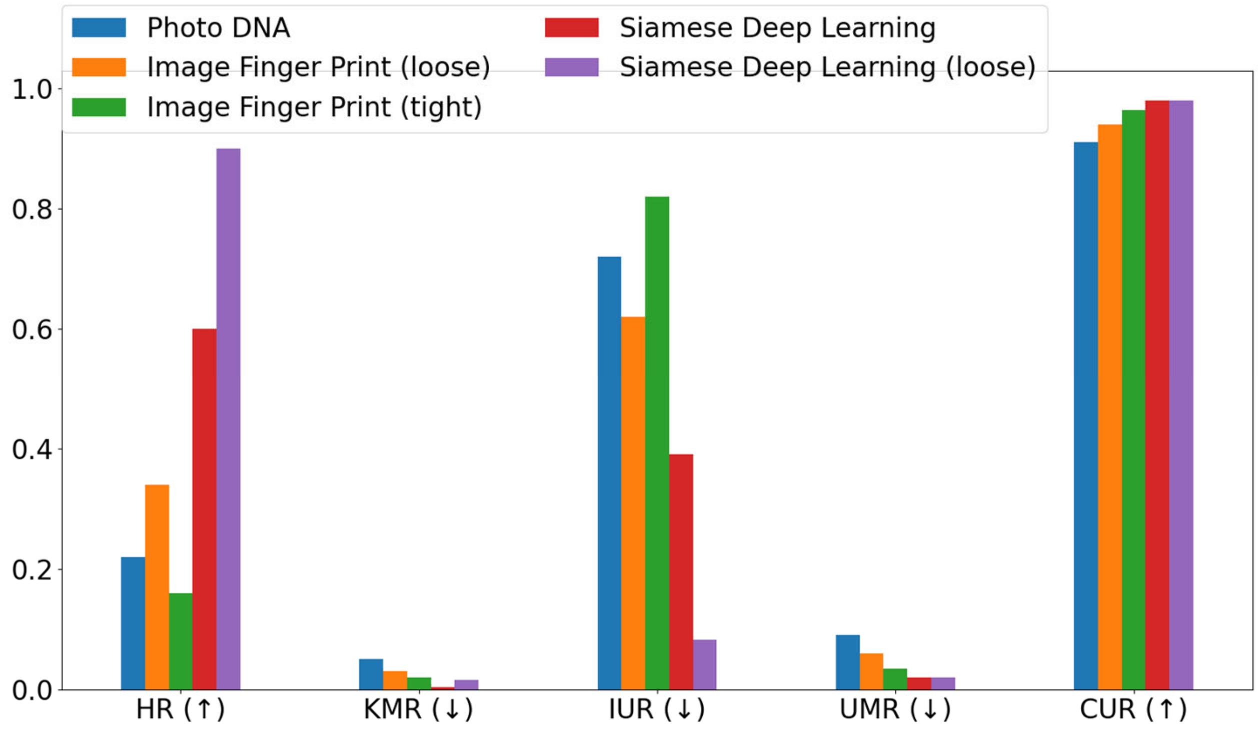
# Unlabeled Malicious Sites




Hits and miss rate of known malicious log in pages without brand labels. We would expect a good classifier to have a high hit rate

# Calibration

- We showed examples where the Siamese Network is the best on *all* metrics. However, it is tunable.
- With a modest 2% increase in the hit rate in the Alexa dataset, we can achieve a 90% hit rate.



# Possible Extensions

- Expanding to other contexts
- Testing for robustness/adversarial perturbation
- Interpreting classification outputs
- Explicitly incorporating logo detection

# Thank you!