Bayes, not Naïve

Provable Security of Website Fingerprinting Defences

Giovanni Cherubin

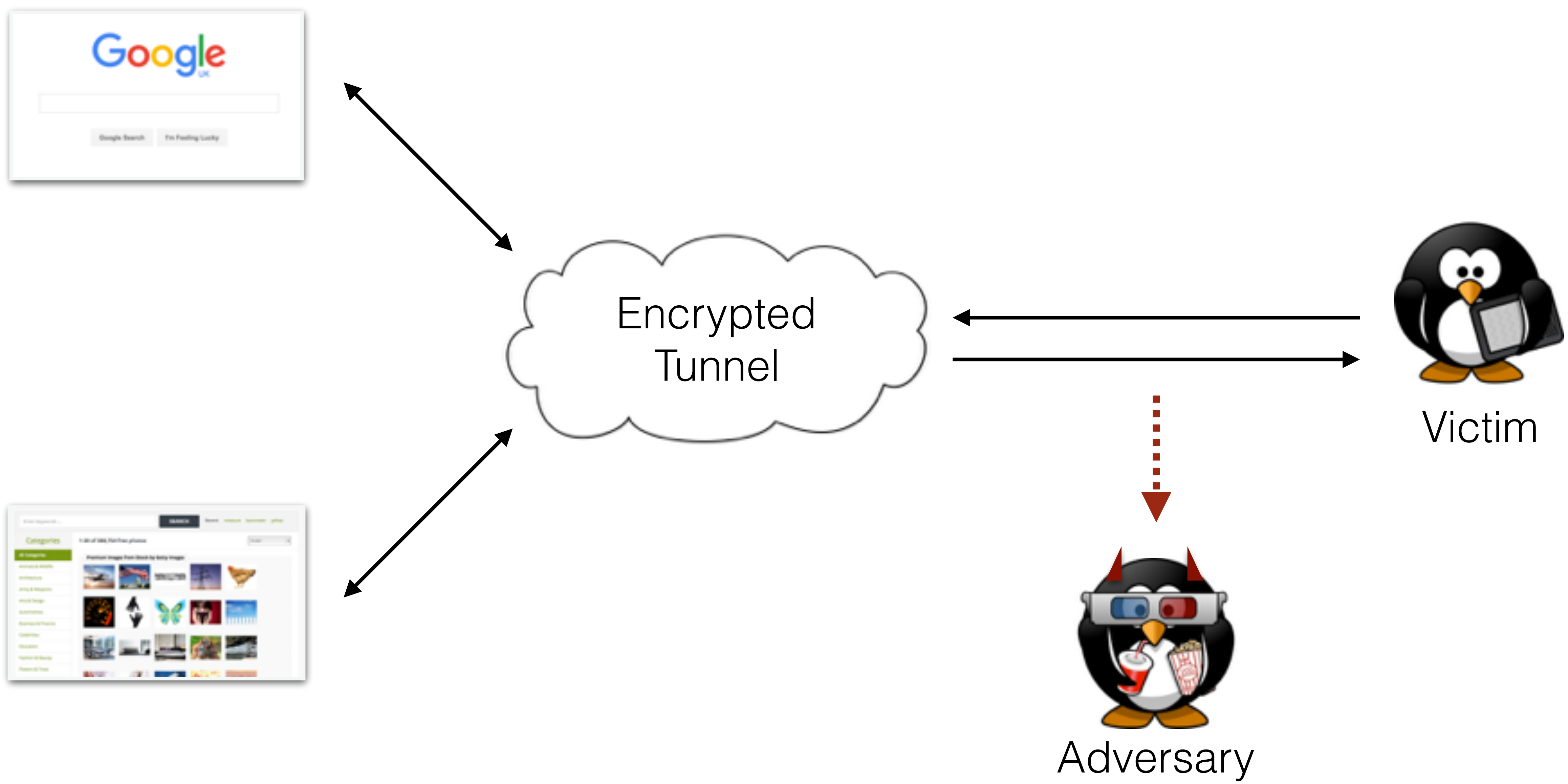
ISG Seminar, Royal Holloway University of London
9 February, 2017

- Background
- Provable bounds by Cai et al.
- Bayes-based bounds, (ε,Φ)-privacy
- Dependence on features

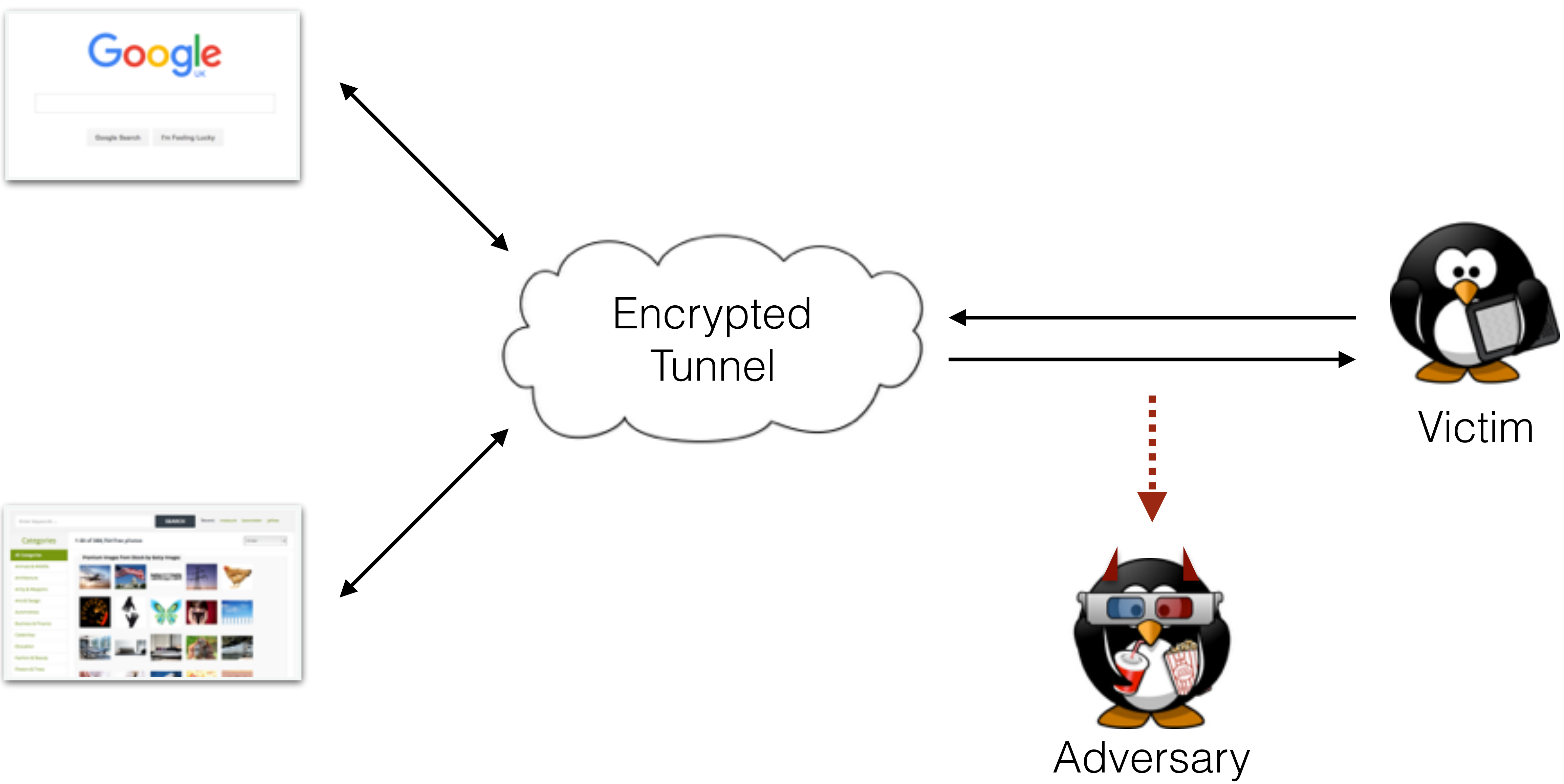
Arms Race

Major attacks and defences up to February 2017

Website Fingerprinting (WF)



Website Fingerprinting (WF)



Threat model

Threat model

Adversary:

Threat model

Adversary:

- **Locally** and **passively** collects network traffic

Threat model

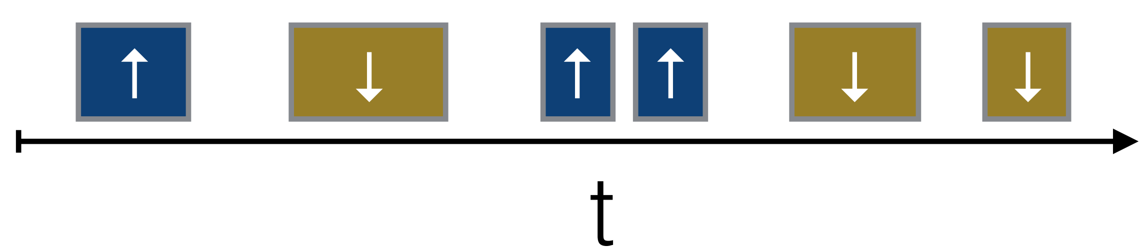
Adversary:

- **Locally** and **passively** collects network traffic
- Does **not** know the **web server's IP** (e.g., victim uses Tor/VPN)

Threat model

Adversary:

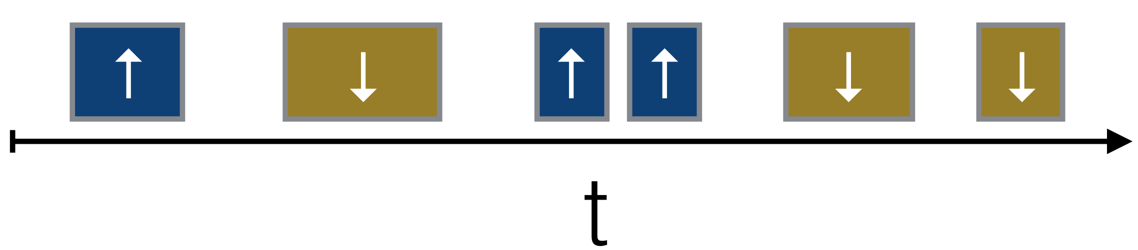
- **Locally** and **passively** collects network traffic
- Does **not** know the **web server's IP** (e.g., victim uses Tor/ VPN)
- Cannot decrypt traffic. Only information he gets from a page load is a **packet sequence**:



Threat model

Adversary:

- **Locally** and **passively** collects network traffic
- Does **not** know the **web server's IP** (e.g., victim uses Tor/VPN)
- Cannot decrypt traffic. Only information he gets from a page load is a **packet sequence**:



- Knows set of web **pages** the user may visit (Closed World)

WF Adversary

 $= (\Phi, \text{Train})$

Feature set

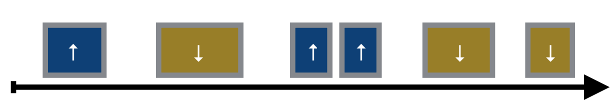
Training algorithm for an ML classifier (e.g., Logistic Regression, Random Forest, SVM)

WF Adversary

 = (Φ, Train)

Training algorithm for an ML classifier (e.g., Logistic Regression, Random Forest, SVM)

Feature set

x = Φ ()

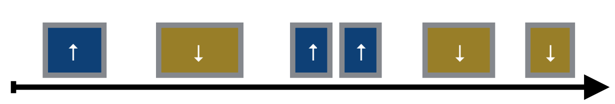
feature object

WF Adversary

$= (\Phi, \text{Train})$

Feature set

Training algorithm for an ML classifier (e.g., Logistic Regression, Random Forest, SVM)

$x = \Phi($  $)$

feature object

$f = \text{Train}((x_1, y_1), \ldots, (x_n, y_n))$

ML classifier

Evaluating WF Attacks

$\mathbf{x}$
test object

$(x_1, y_1), \ldots, (x_n, y_n)$
training set

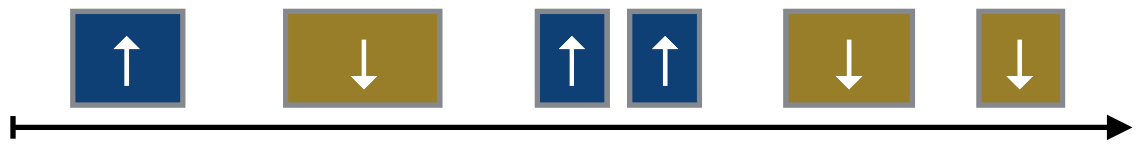
$f(x)$

$R^f = \Pr(f(x) \neq y)$

Implicit: $x_i = \Phi(p_i)$

Probabilistic & Deterministic Defences

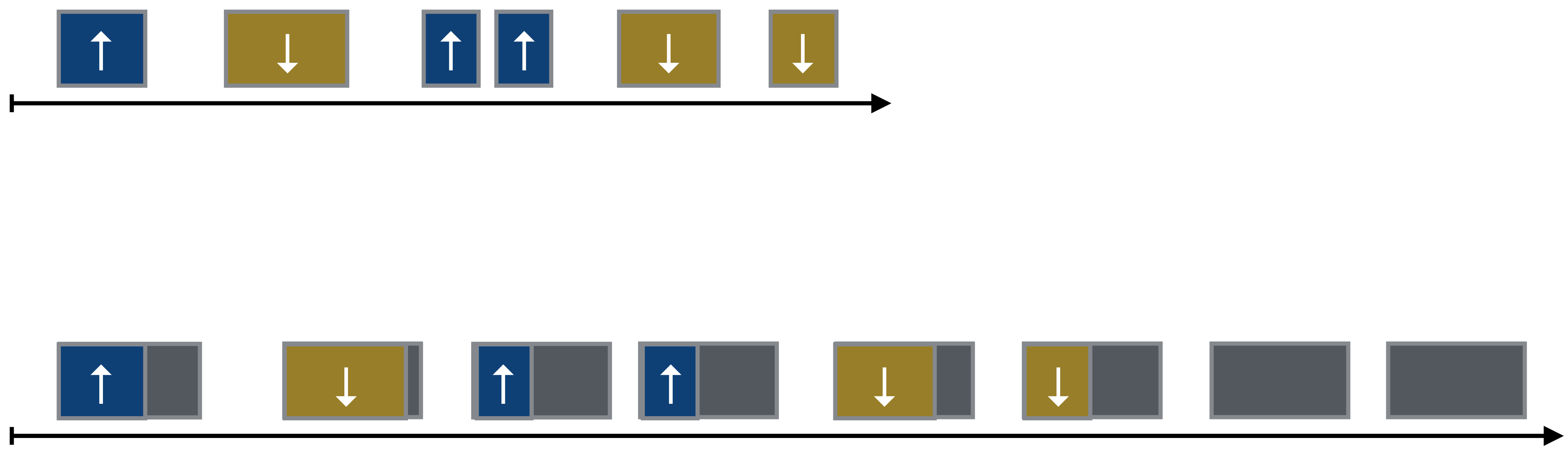
Original



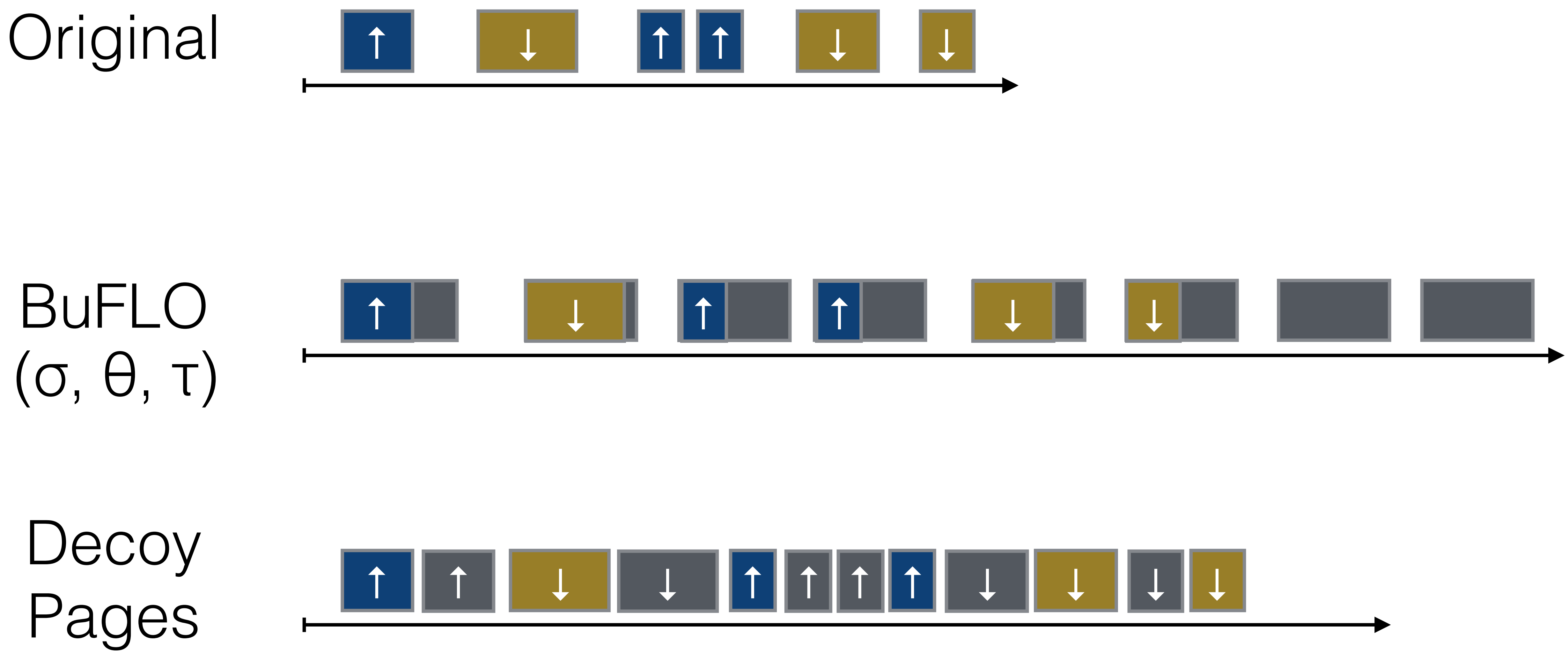
Probabilistic & Deterministic Defences

Original

BuFLO
(σ, θ, τ)



Probabilistic & Deterministic Defences



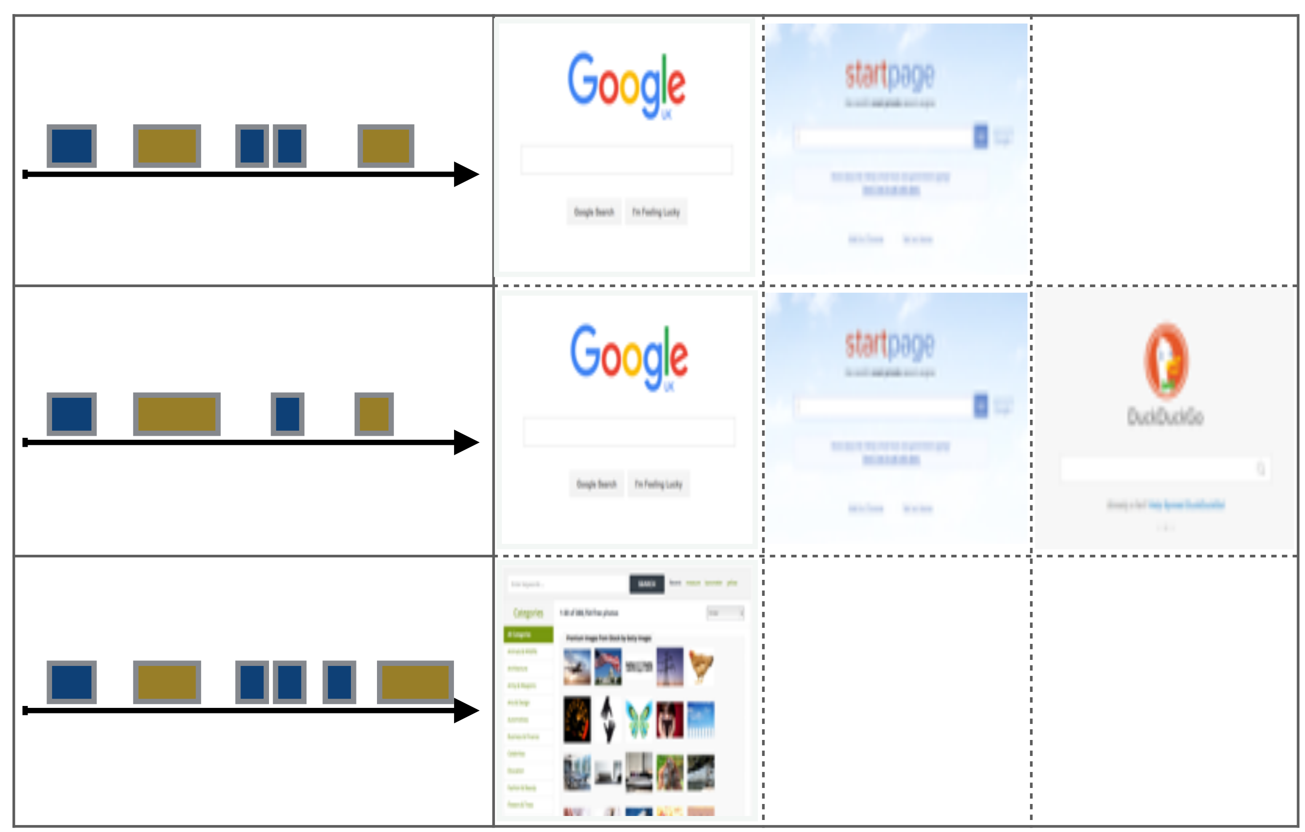
- Background
- **Provable bounds by Cai et al.**
- Bayes-based bounds, (ε,Φ)-privacy
- Dependence on features

“Lookup-Table” Approach

(Cai et al., ’14)

Idealised Adversary: knows exactly what packet sequences each web page may generate. Counts the collisions.

Lookup table



“Lookup-Table” bound

- Highly affected by **noise**
- Needs to be computed on **partial** data
- Only **deterministic** defences

- Background
- Provable bounds by Cai et al.
- **Bayes-based bounds, (ε,Φ)-privacy**
- Dependence on features

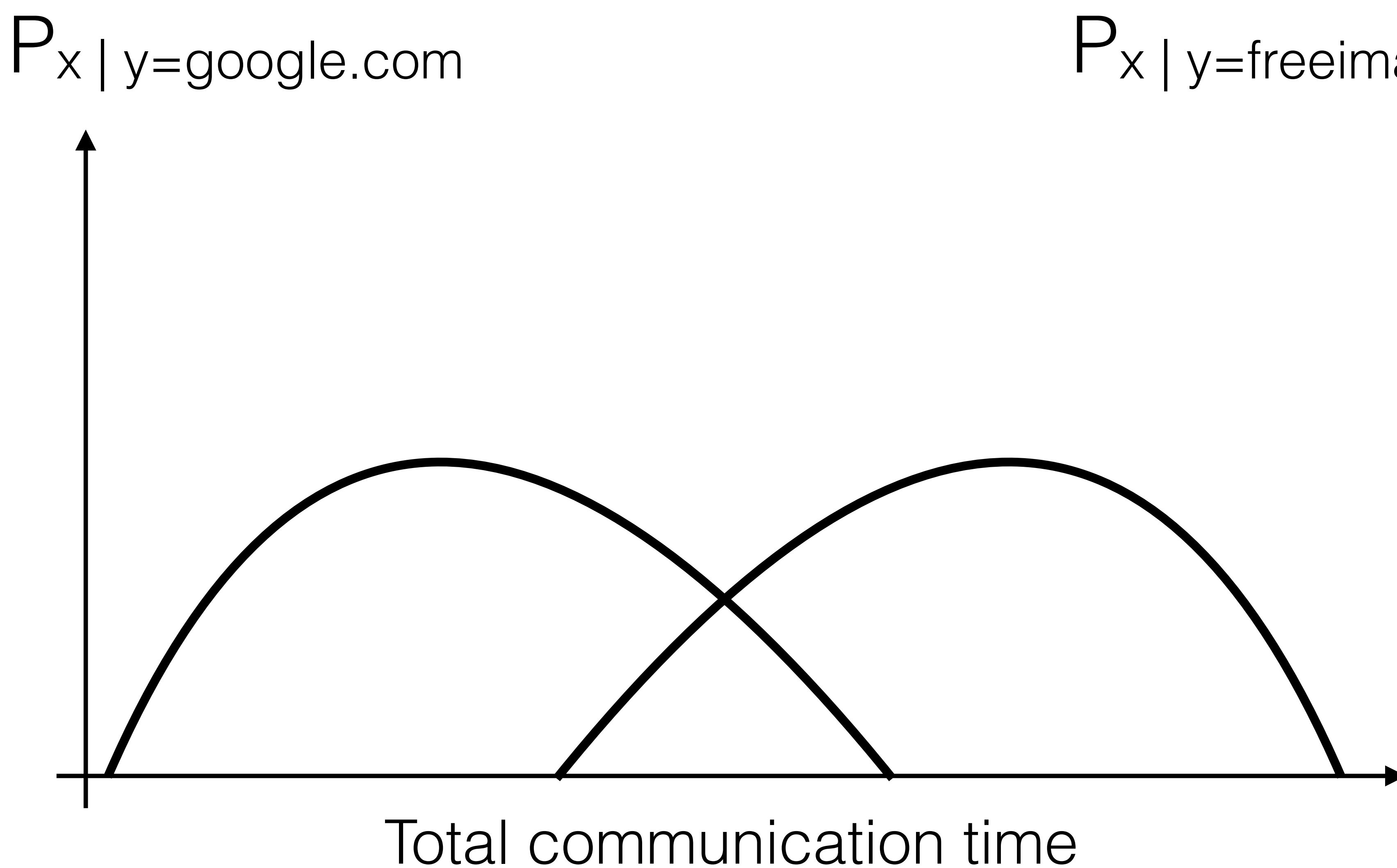
“Bayes error” approach

 = (Φ, Train)

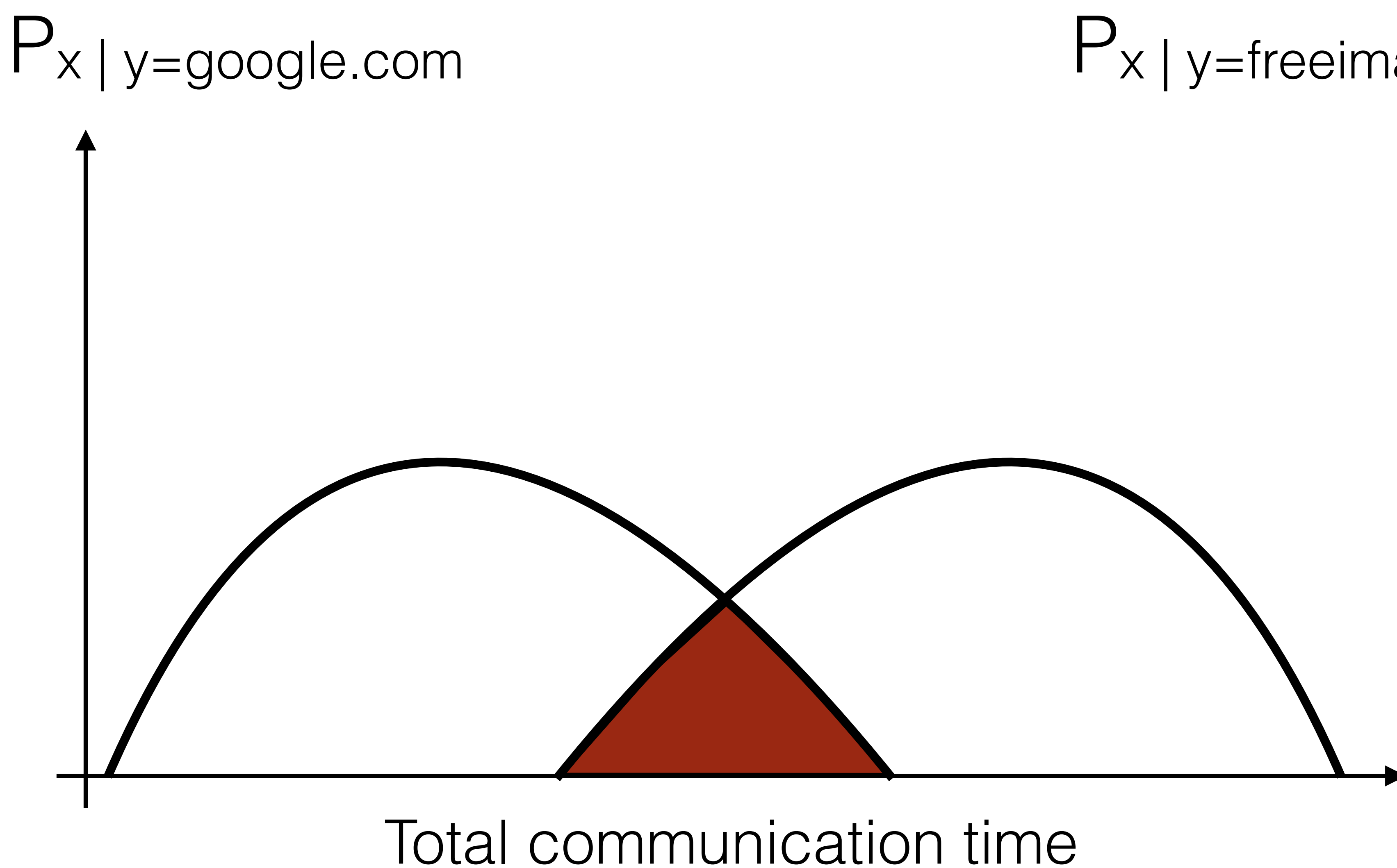
Distinguishing Web Pages

$P_{x \mid y=\text{google.com}}$ $P_{x \mid y=\text{freeimages.com}}$

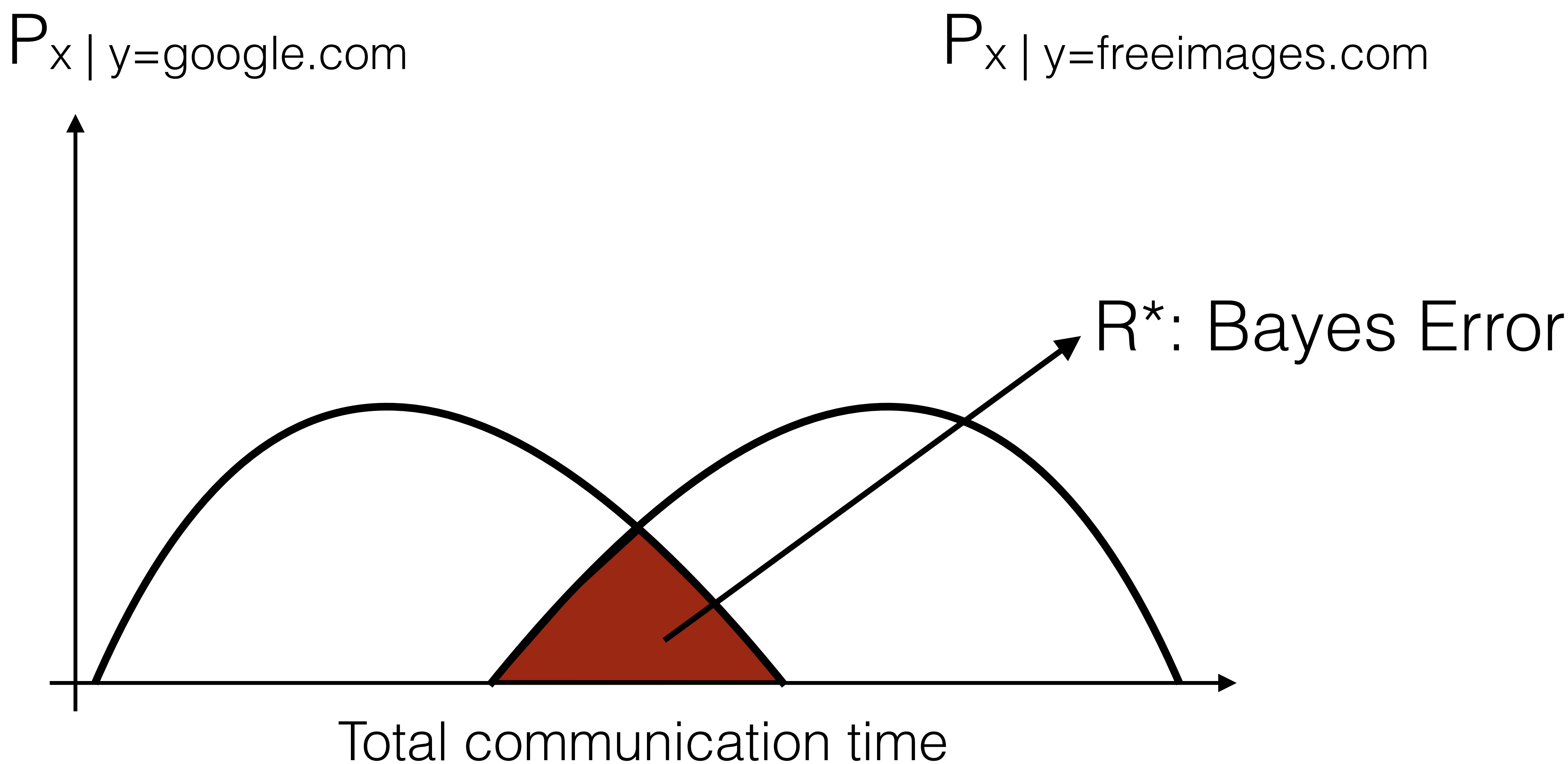
Distinguishing Web Pages



Distinguishing Web Pages



Distinguishing Web Pages



"Bayes error" approach

$$f = \text{Train}\left(\underset{\text{training set}}{(x_1, y_1), \ldots, (x_n, y_n)}\right)$$

$$\Downarrow$$

$$R^f = \Pr\left(f\left(\underset{\text{test object}}{x_{n+1}}\right) \neq y_{n+1}\right)$$

Implicit: $x_i = \Phi(p_i)$

"Bayes error" approach

$$f = \text{Train}\left(\underset{\text{training set}}{(x_1, y_1), \ldots, (x_n, y_n)}\right)$$

$$R^f = \Pr\left(f\left(\underset{\text{test object}}{x_{n+1}}\right) \neq y_{n+1}\right)$$

$$R^f \geq R^*$$

Implicit: $x_i = \Phi(p_i)$

“Bayes error” approach

$$f = \text{Train}\left(\underset{\text{training set}}{(x_1, y_1), \ldots, (x_n, y_n)}\right)$$

$$R^f = \Pr\left(f\left(\underset{\text{test object}}{x_{n+1}}\right) \neq y_{n+1}\right)$$

$$R^f \geq R^* \geq \hat{\mathbf{R}}^*$$

Implicit: $x_i = \Phi(p_i)$

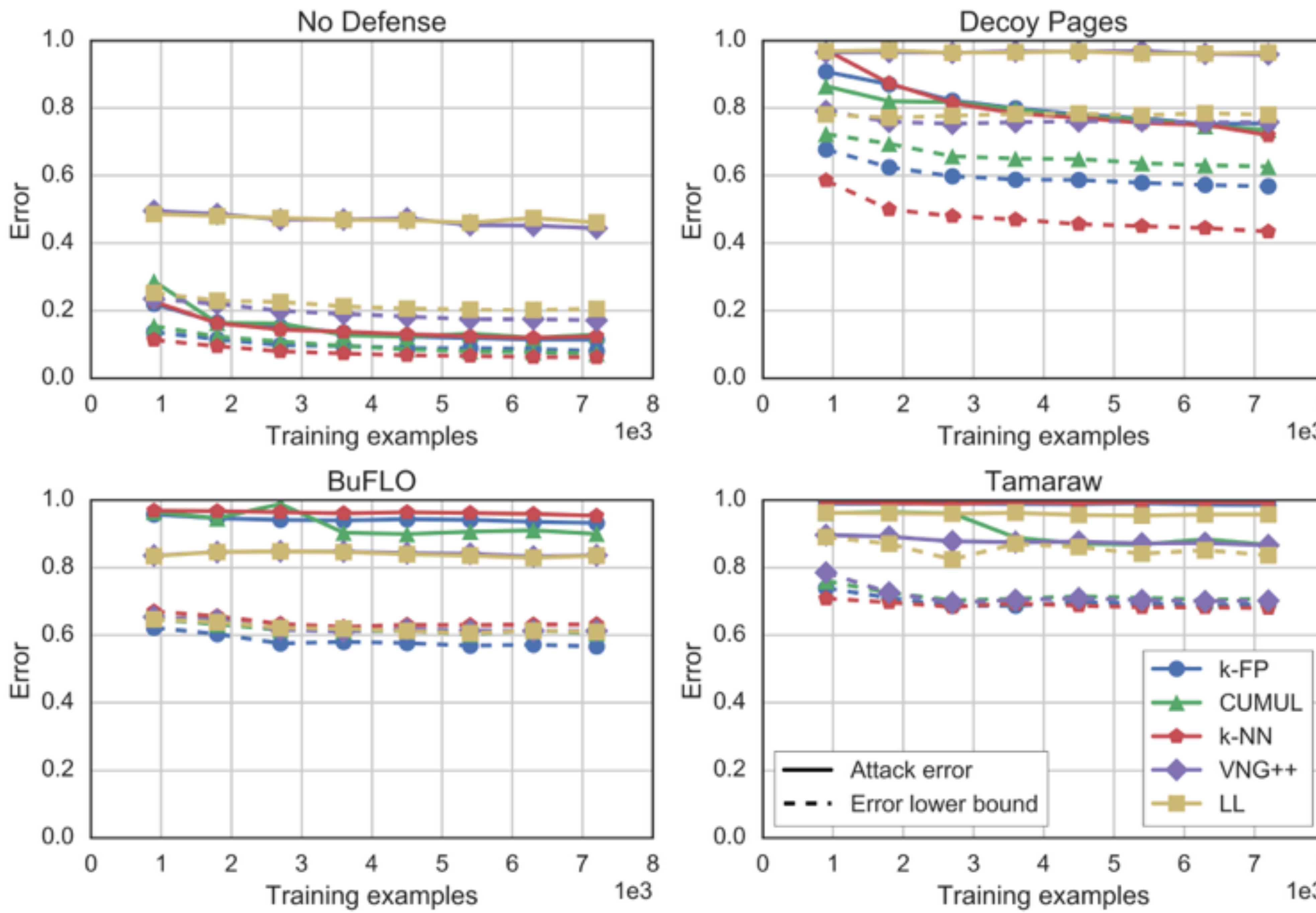
Bayes Error Estimate

(Cover & Hart, '67)

Asymptotically,

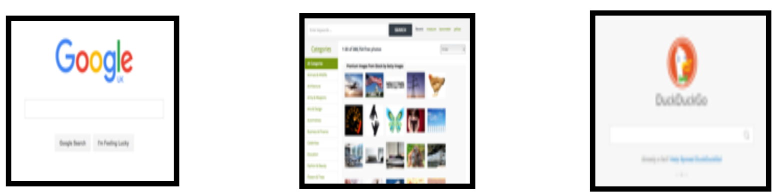
$$R^* \geq \frac{L-1}{L}\left(1 - \sqrt{1 - \frac{L}{L-1}R^{NN}}\right)$$

NN-based estimate



(ε, Φ)-privacy

Problem An error estimate $\hat{R}^*$ alone does not convey information about the setting



?

$R^G = 2/3$

?

$R^G = 1/2$

Define metric (1 - Adv):

$$\varepsilon = \hat{R}^* / R^G$$

- Background
- Provable bounds by Cai et al.
- Bayes-based bounds, (ε,Φ)-privacy
- **Dependence on features**

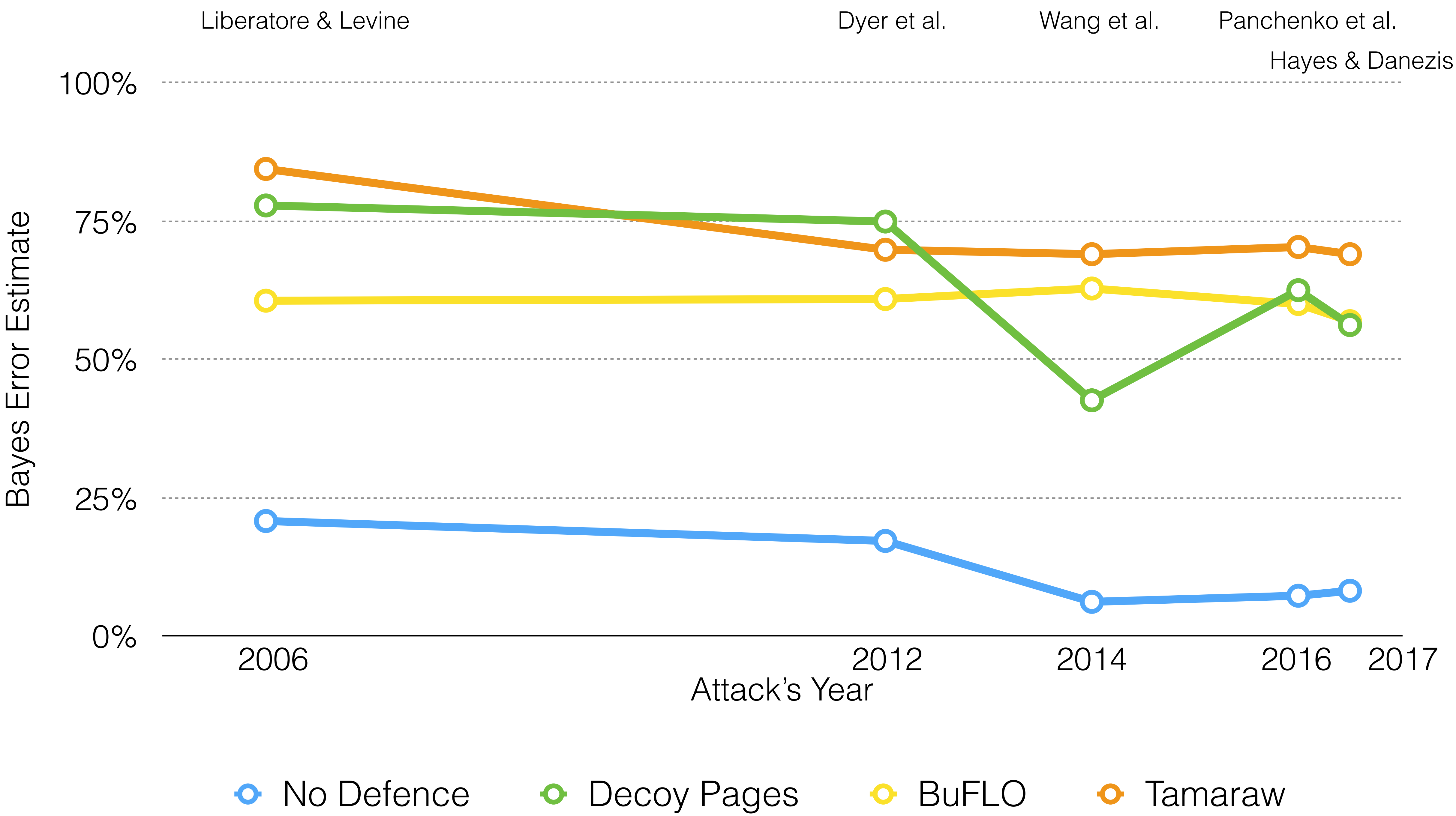
Do We Need Features?

Theorem Using the full original data performs no worse than using any transformation (i.e., feature) of the original data *asymptotically.*

Problem *Curse of dimensionality* and *difficult separability* in original space.

(How much)

Did Feature Sets Improve?



Features

Determine an **efficient** set of features Φ:

- low computational-memory *complexity* to extract objects
- good *accuracy* w.r.t. the optimal one: balance dimensionality and information

$$R^{\Phi} \leq R^{\Phi'} + \varepsilon$$

TL;DL

- WF attacks **too long** without security proofs
- Bound based on **lookup-table** suffer from **noise**
- **R* estimate** bounds an adversary A = (Φ, ·)
- R* can be used to evaluate features
- Improving **Φ** is becoming more difficult

Future Directions

Efficient feature set:

- how to show that a feature set is efficient?
- can we construct one automatically?

New estimates of R*:

- tighter bounds
- weaker assumptions on data

Other applications of the method

Bayes, not Naïve

Provable Security of Website Fingerprinting Defences

Giovanni Cherubin

ISG Seminar, Royal Holloway University of London
9 February, 2017

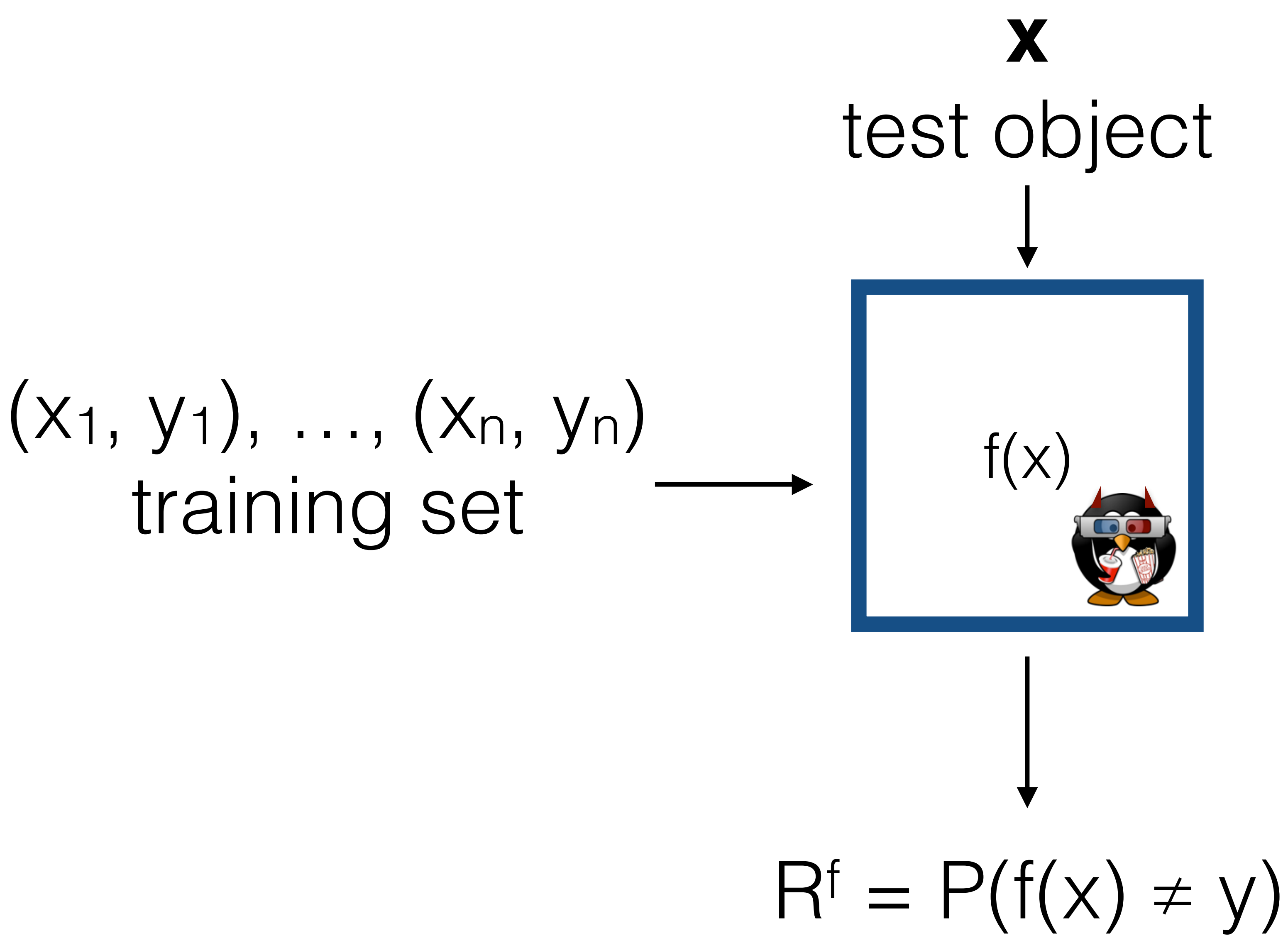
Comparision with Cai et al.

Defence	R* estimate	Cai et al.	Cai et al. (full information)
BuFLO	57%	53%	19%
Tamaraw	69%	91%	11%

(ε,Φ)-privacy

Defence	(ε,Φ)-privacy	Time OH	Packet OH
No Defence	(0.06, k-NN)	0%	0%
Decoy Pages	(0.43, k-NN)	29%	98%
BuFLO	(0.58, k-FP)	24%	11%
Tamaraw	(0.70, k-NN)	334%	161%

Other applications



Bayes, not Naïve

Provable Security of Website Fingerprinting Defences

Giovanni Cherubin

ISG Seminar
9 February, 2017