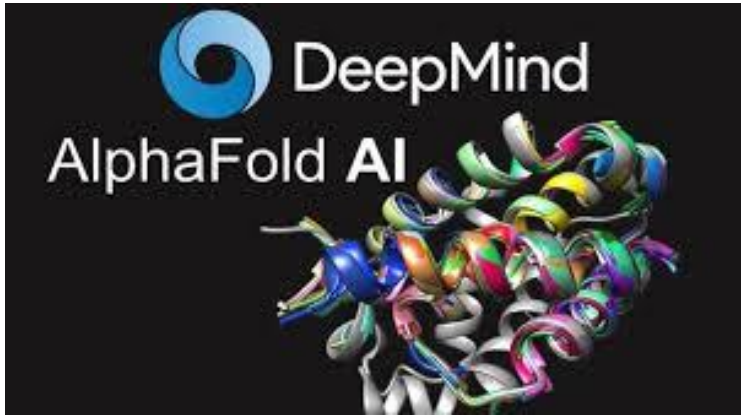
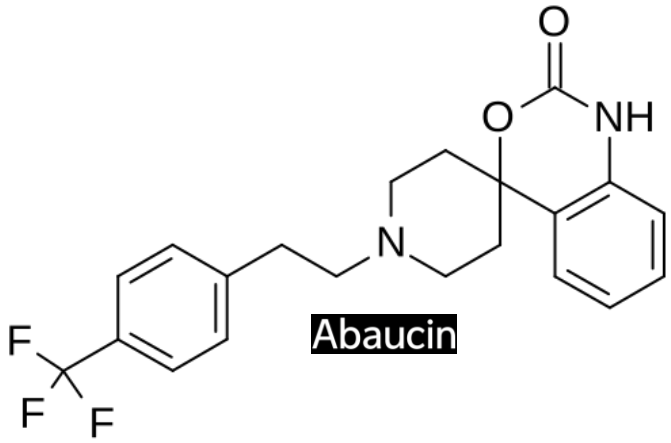
Cryptography and AI: Challenges and Opportunities

Vinod Vaikuntanathan

Based on joint works with **Andrej Bogdanov**, **Shafi Goldwasser**, **Alon Rosen**, **Jonathan Shafer**, **Neekon Vafa**, and **Or Zamir**. Some slides are borrowed from these generous people.

AI is here…





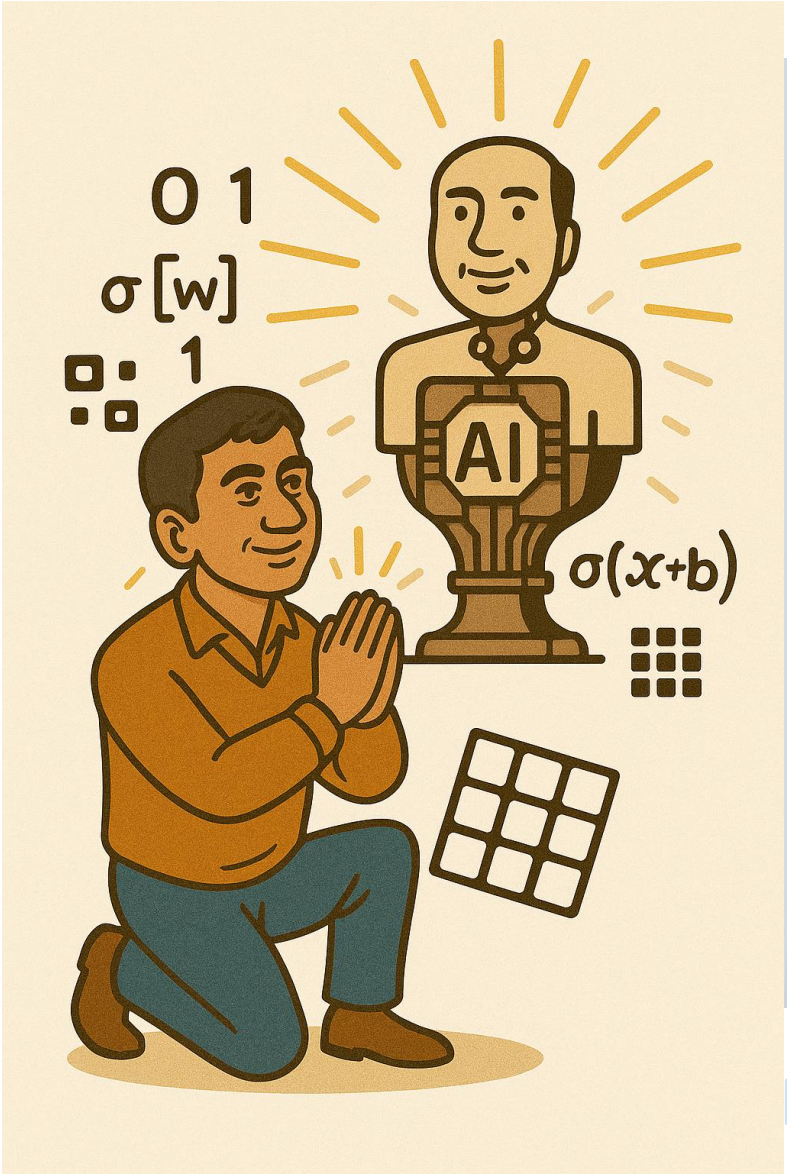
Scientific Discovery

Programming

Mathematical Problem Solving

AI is here…

Scientific Discovery



Mathematical Problem Solving

https://en.wikipedia.org/wiki/Stochastic_parrot

https://ai-2027.com

(Some) AI Challenges

Trustworthy AI

- **Hallucination:** *how do I know the model outputs are correct?*
- **Robustness:** *will the model remain good under distribution shift?*
- **(Adversarial) Control:** *susceptible to undetectable manipulation?*
- **Alignment:** *are the model's goals aligned to ours?*

- **Resource Consumption:** *can we lower the giant compute use?*

My Thesis:

Cryptography and Cryptographers have a Role to Play

Adversarial Thinking

The Omnipresent Adversary

Definitions

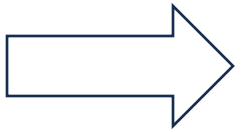
What access does the adversary have?

What are her goals?

What measures her success?

Turning Hardness into Usefulness

Hard Problems



Useful Systems

Proofs via Reductions (or the "win-win paradigm"):

CAPTCHA:
Using Hard AI Problems For Security

Luis von Ahn[1], Manuel Blum[1], Nicholas J. Hopper[1], and John Langford[2]

[1] Computer Science Dept., Carnegie Mellon University, Pittsburgh PA 15213, USA
[2] IBM T.J. Watson Research Center, Yorktown Heights NY 10598, USA

Verification of a human in the loop
or
Identification via the Turing Test*

Moni Naor †

September 13th, 1996

… but we may need to think differently

Crypto

theory first

adversarial

maximalistic

ML/AI

empirical

optimistic

pragmatic

Different models, Different goals, Different adversaries.

Need new ideas, new tools, new hard problems.

This Talk

Robust Embeddings

Backdoors

Alignment

Crypto to Speed up (ML) Algorithms

What I won't get to talk about

Defining Generative Models: given n samples from a distribution D, want to (learn to) generate more samples from D. What does that mean?

[Kleinberg-Mullainathan'24 "Language Generation in the Limit"]

Watermarking: many wonderful results. Even more open problems, e.g. do unremovable watermarks even exist?

[Christ-Gunn-Zamir, Barak et al.'24 "Watermarks in the Sand…", Sahai et al.'25 "Sandcastles in the storm…"]

Verification: can models prove their correctness?

[Amit, Goldwasser, Paradise, Rothblum'25 "Self-proving models…"]

Privacy, Secure Computation: secure inference and/or training

Making ML models "forget": Machine unlearning.

This Talk

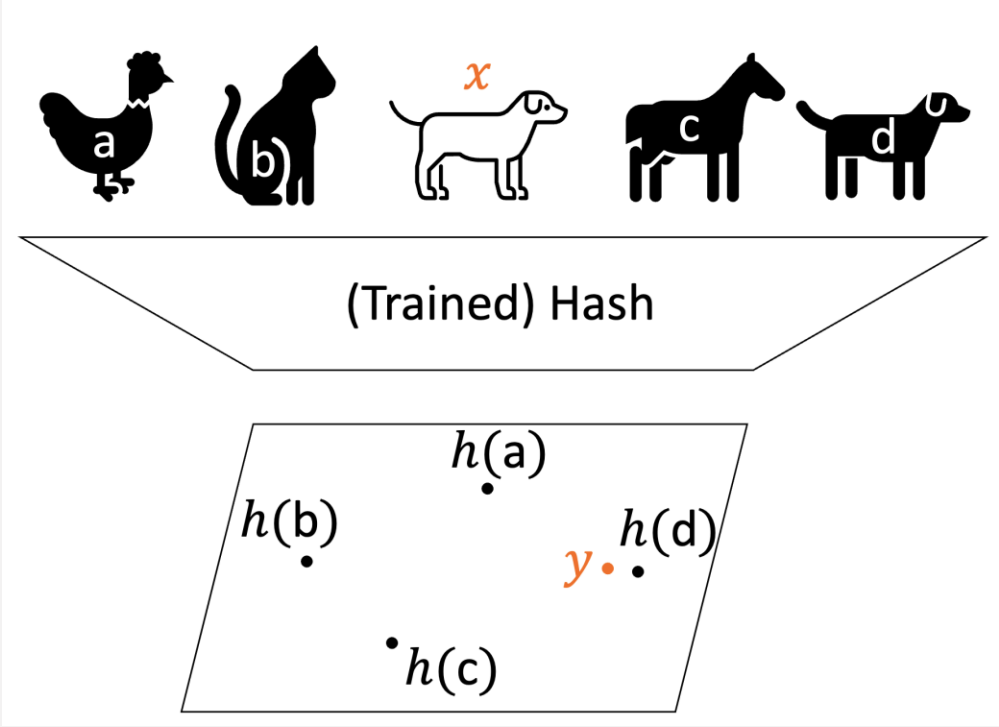
Robust Embeddings

Backdoors

Alignment

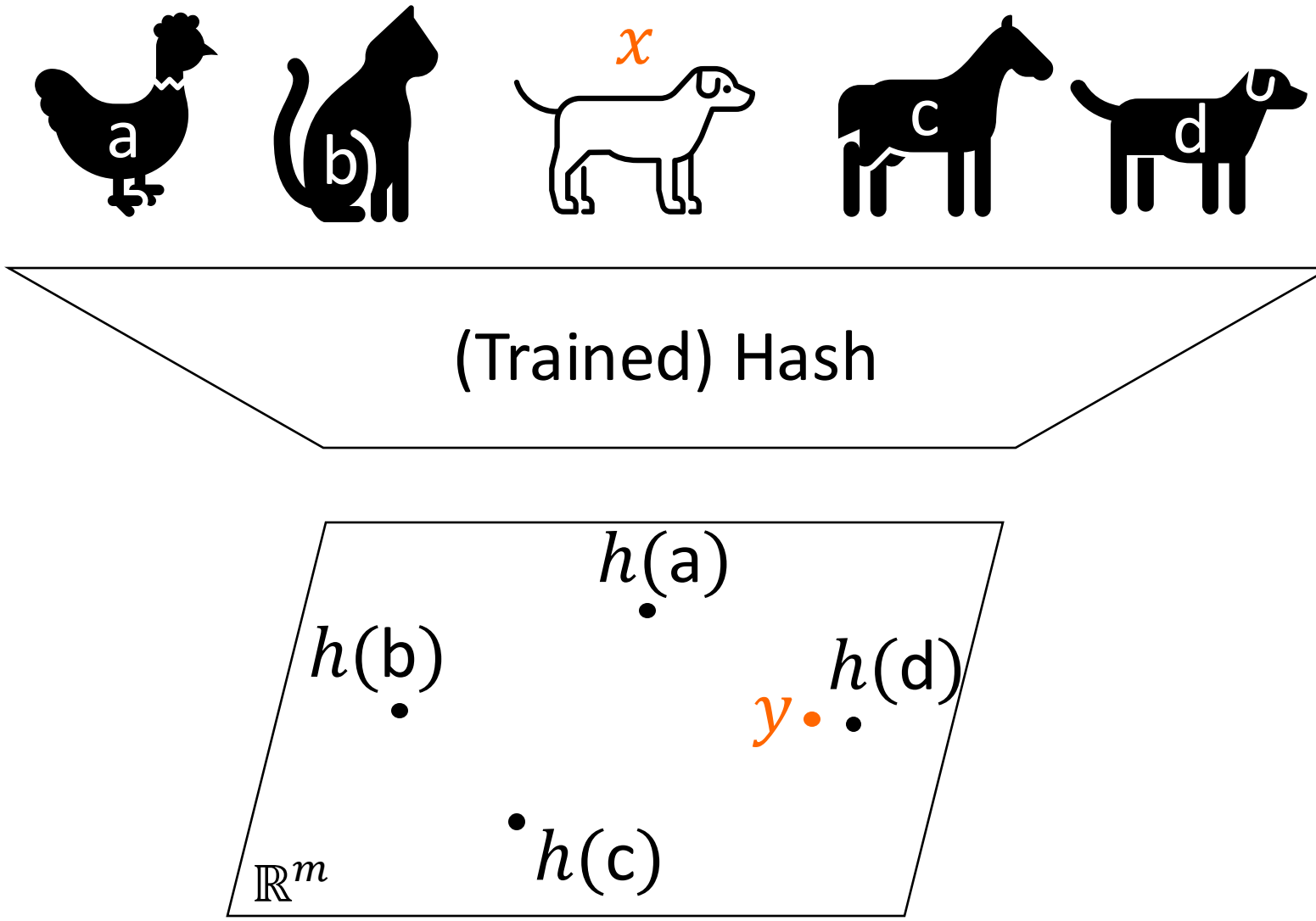
Crypto to Speed up (ML) Algorithms

Robust ML Embeddings



Courtesy: Andrej, Alon and Neekon for some slides.

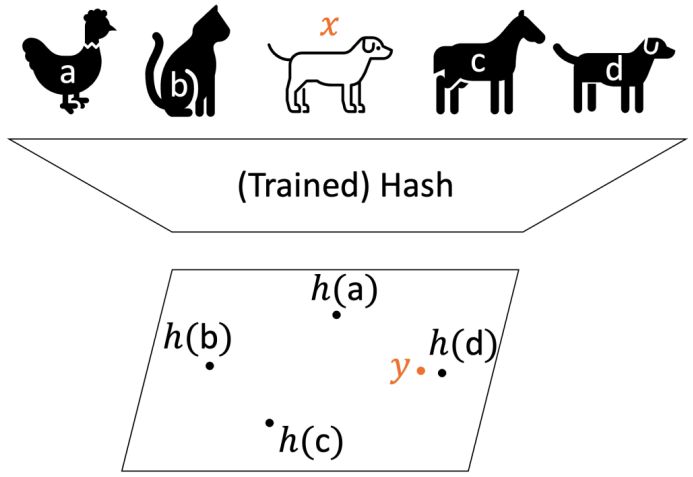
ML Embeddings



Semantic Similarity

Geometric Closeness

ML Embeddings



Word2Vec [Mikolov et al. 2013]

Early neural system that mapped words to vectors.

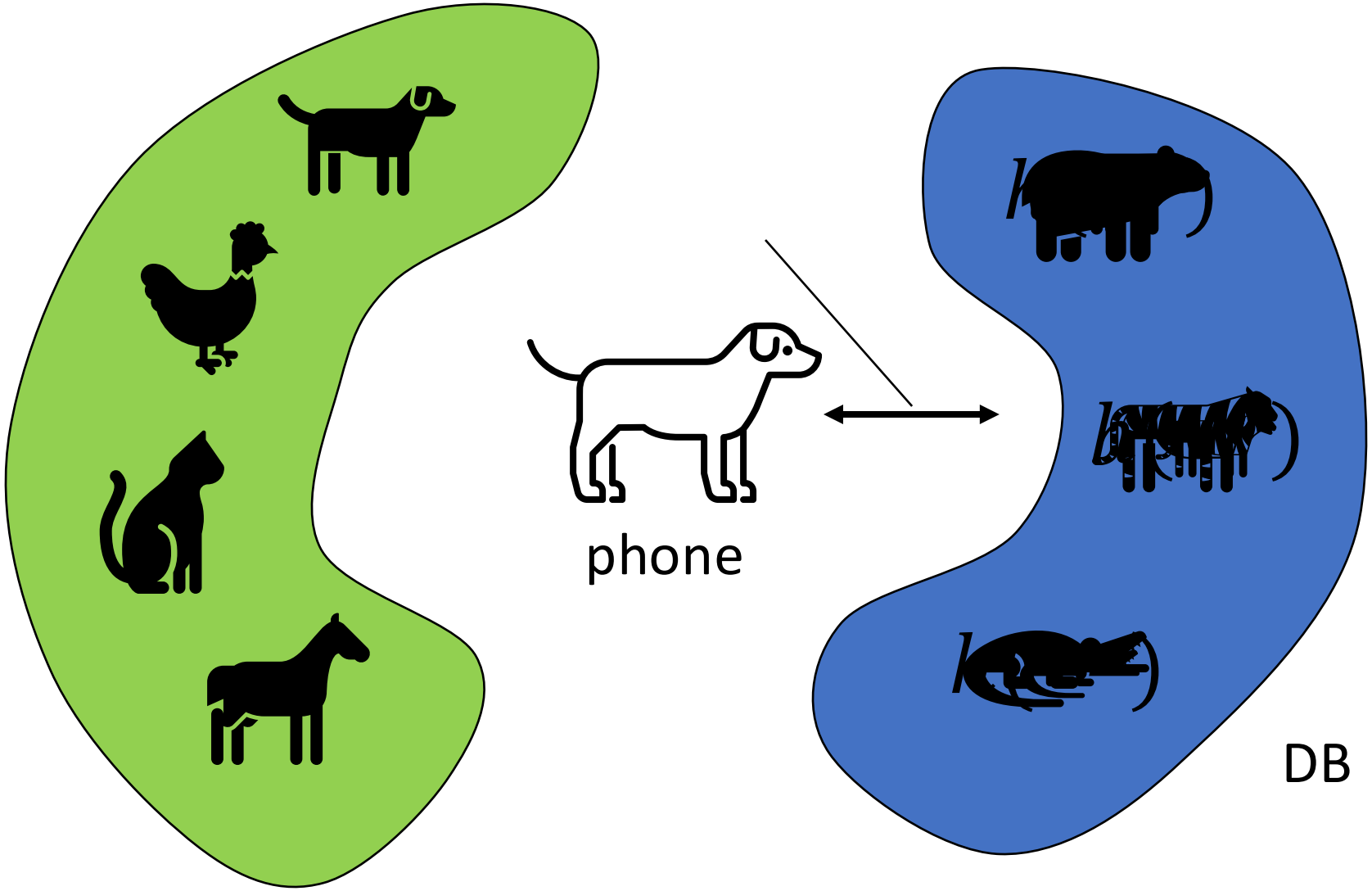
Vision Transformers [Dosovitsky et al. 2021]

Compressing map from e.g. 224 X 224 RGB image to a 768-dimensional vector (with 32-bit precision)

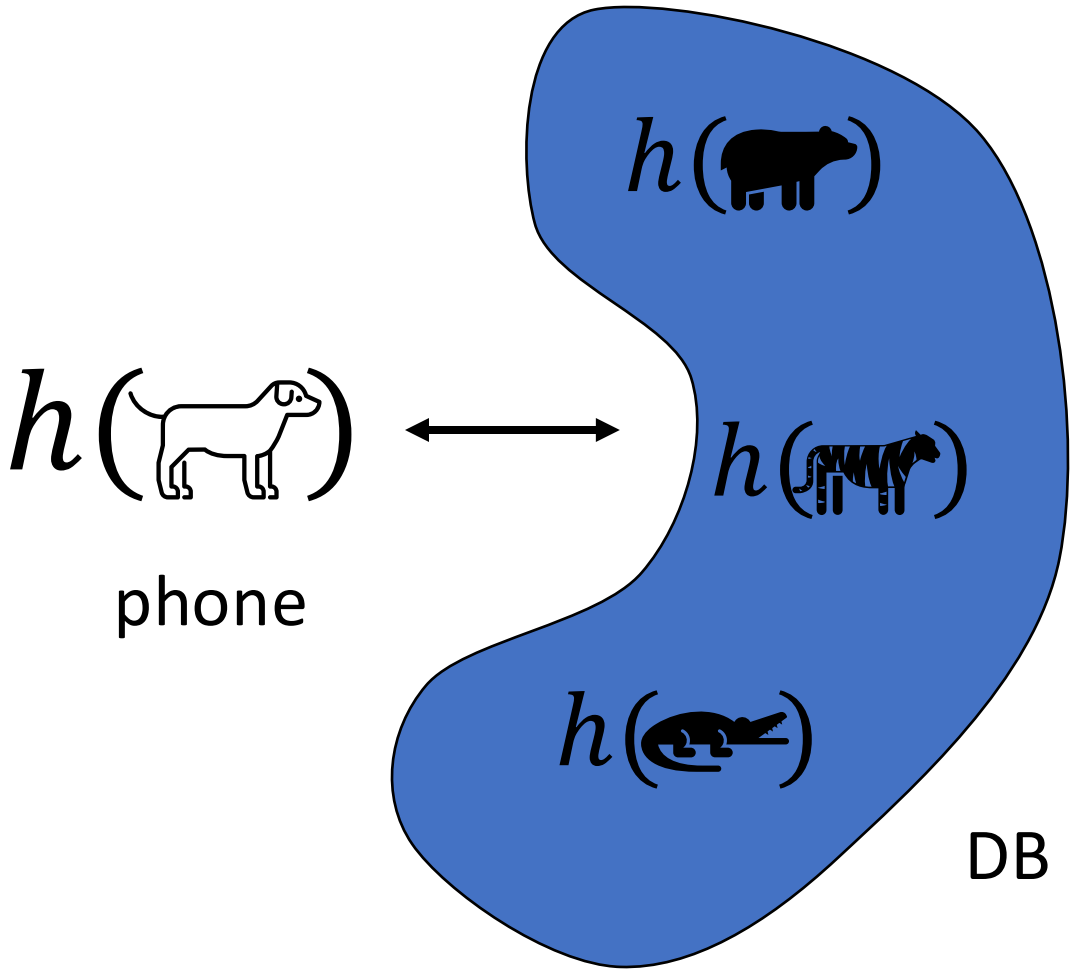
CLIP [Li et al. 2016, Radford et al. 2021]

Multimodal: connect text and image embeddings!

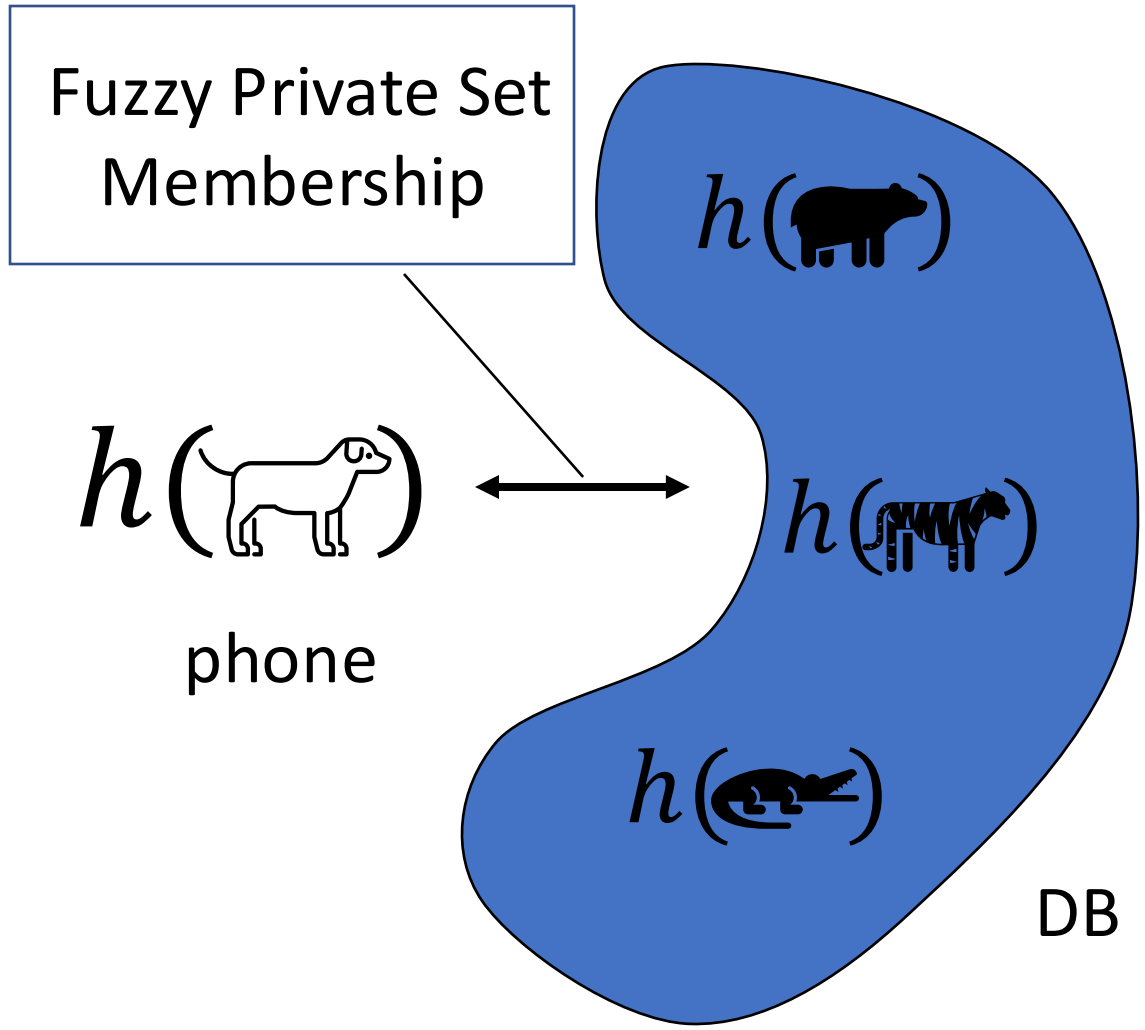
Apple NeuralHash [2021]



Apple NeuralHash [2021]



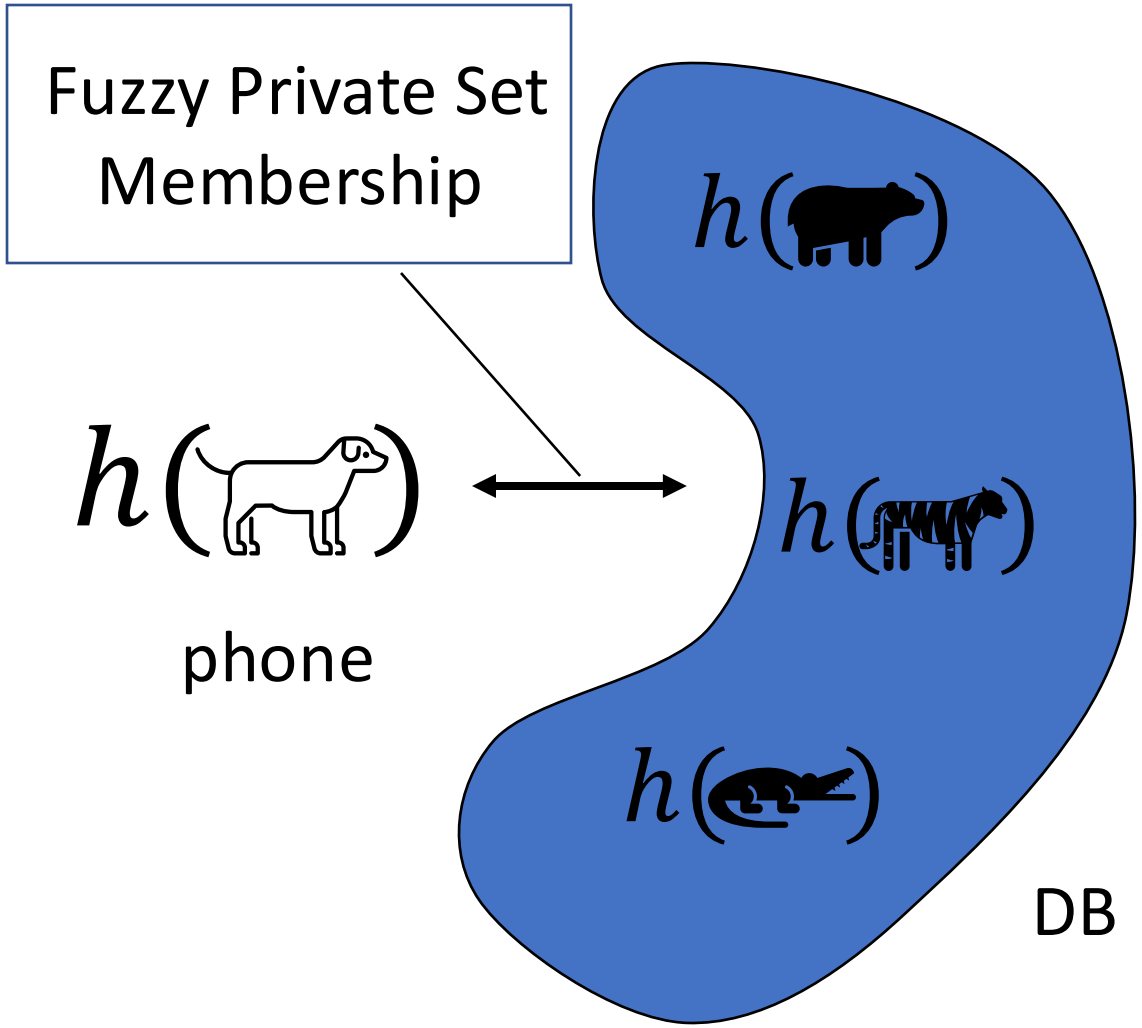
Apple NeuralHash [2021]



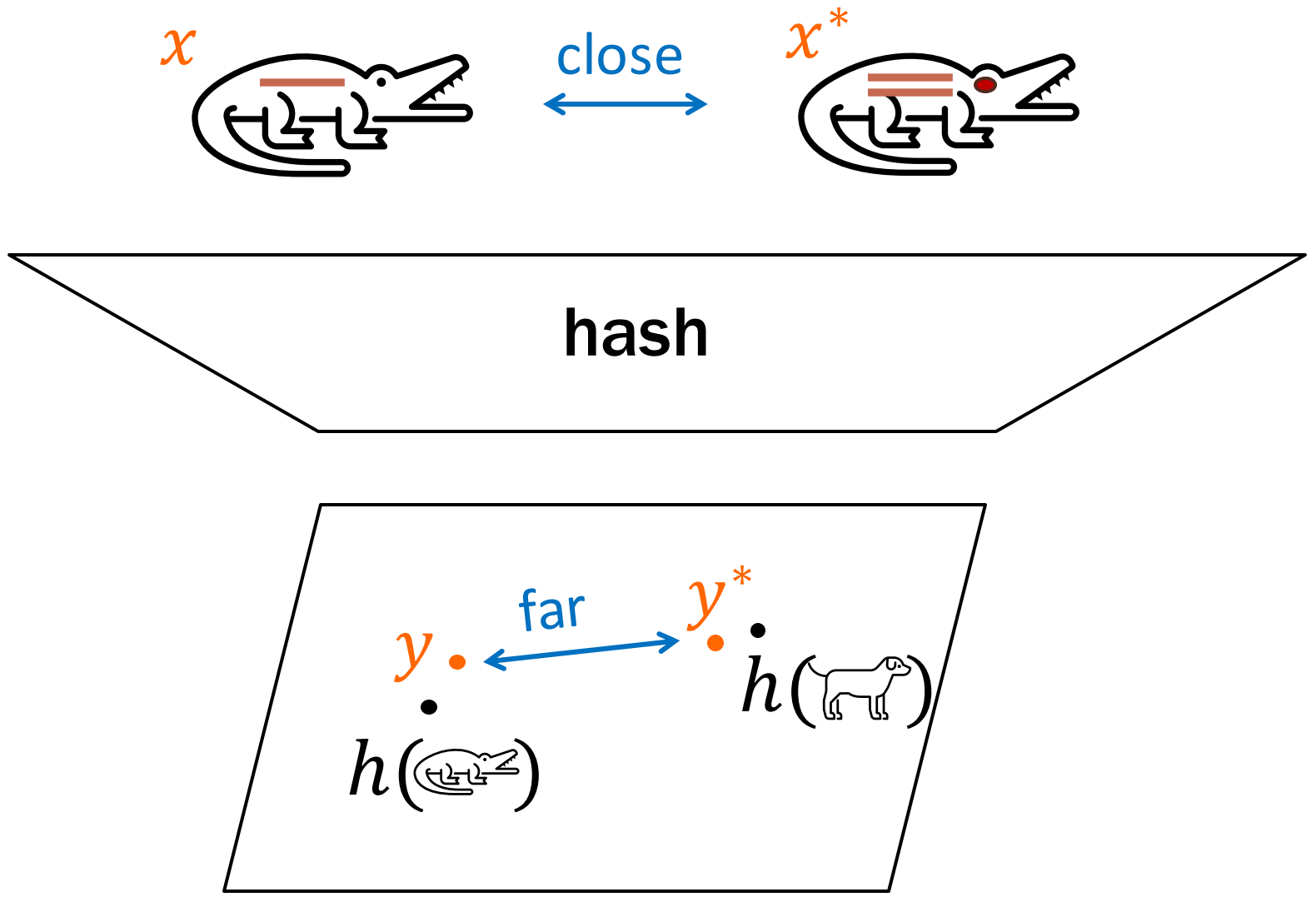
Apple NeuralHash [2021]

[Bhatia, Meng '22]

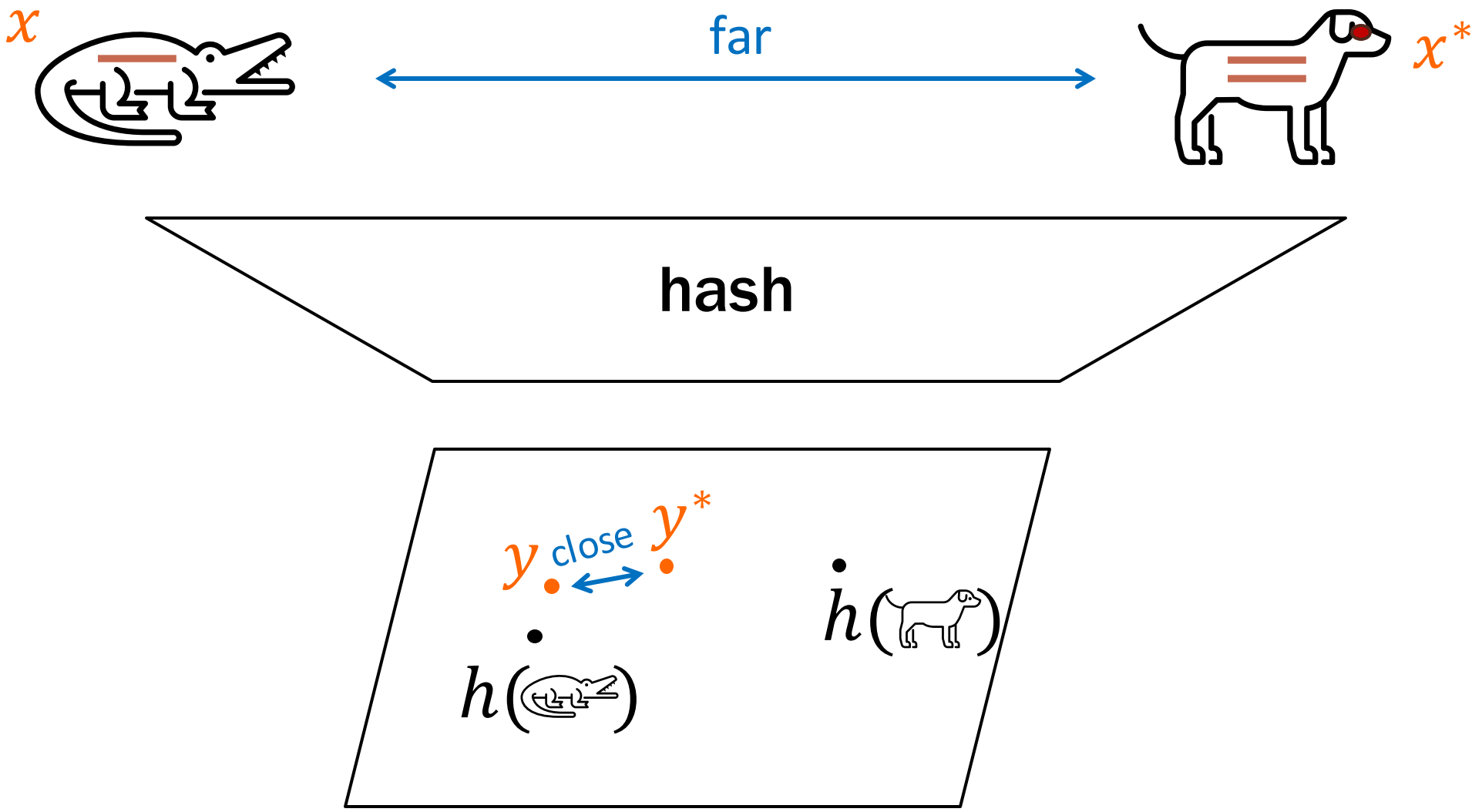
Apple's NEURALHASH is one such system that aims to detect the presence of illegal content on users' devices without compromising consumer privacy. We make the surprising discovery that NEURALHASH is *approximately linear*, which inspires the development of novel black-box attacks that can (i) evade detection of "illegal" images, (ii) generate near-collisions, and (iii) leak information about hashed images, all without access to model parameters.



Adversarial Expansion (Evasion)



Adversarial Contraction (Framing)



Euclidean Embeddings

Lemma (**Johnson-Lindenstrauss'84, Indyk-Motwani'99**): Fix $0 < \varepsilon < 1$ and let $m = \Omega(\frac{\lambda}{\varepsilon^2})$. Let $\boldsymbol{h(x) = h_A(x) = \frac{1}{\sqrt{m}} Ax}$ where $A \sim N(0,1)^{m \times n}$ is a Gaussian matrix. For every $x, y \in \mathbb{R}^n$: ← *chosen independent of A*

$$\Pr[||h(x) - h(y)|| \notin (1 \pm \varepsilon)||x - y||] \leq 2^{-\lambda}$$

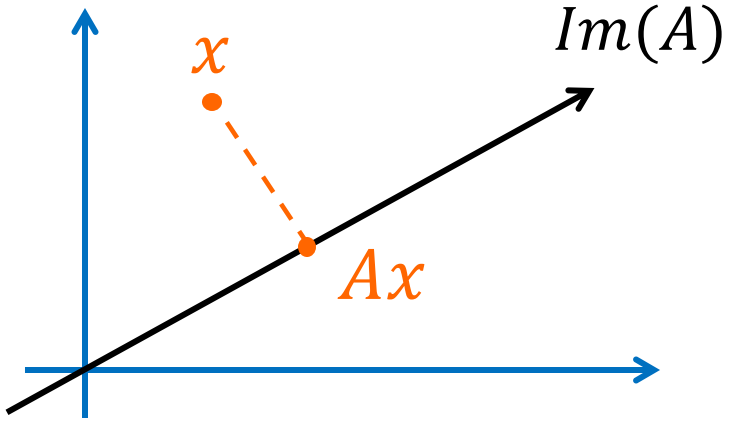
Adaptive Robustness of JL

Lemma (**Johnson-Lindenstrauss'84, Indyk-Motwani'99**): Fix $0 < \varepsilon < 1$ and let $m = \Omega(\frac{\lambda}{\varepsilon^2})$. Let $\boldsymbol{h(x) = h_A(x) = \frac{1}{\sqrt{m}} Ax}$ where $A \sim N(0,1)^{m \times n}$ is a Gaussian matrix. For every $x, y \in \mathbb{R}^n$: ← *chosen independent of A*

$$\Pr[||h(x) - h(y)|| \notin (1 \pm \varepsilon)||x - y||] \leq 2^{-\lambda}$$

Given A: easy! find a vector in the kernel of A.

Even with super-weak "oracle access" to $\boldsymbol{h_A}$: [Hardt-Woodruff'13] showed how to recover a "good enough" A' and run the kernel attack.



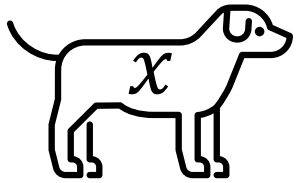
Robust Locality-Sensitive Hash

[Boyle-Lavigne-**V**.'19]

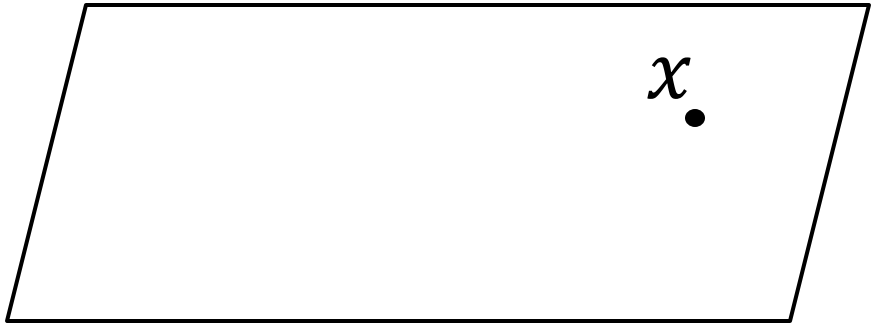
A hash function family $\{h_K: \mathbb{R}^n \to \mathbb{R}^m\}$ is a robust (Euclidean) LSH if:

- **Compressing:** The output length (in bits) is smaller than the input length.
- **α-expanding (for $\alpha > 1$):** given K, no p.p.t. adversary can find $x, y \in \mathbb{R}^n$ s.t. $||h_K(x) - h_K(y)|| > \alpha \cdot ||x - y||$.
- **β-contracting (for $\beta < 1$):** given K, no p.p.t. adversary can find $x, y \in \mathbb{R}^n$ s.t. $\left|\left|h_K(x) - h_K(y)\right|\right| < \beta \cdot ||x - y||$.

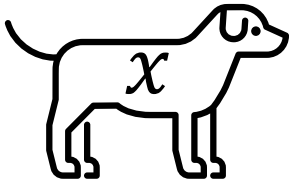
(Computational) Distortion: α/β, ideally close to 1.


x


ML trained hash


x

New Paradigm: ML + Crypto



ML

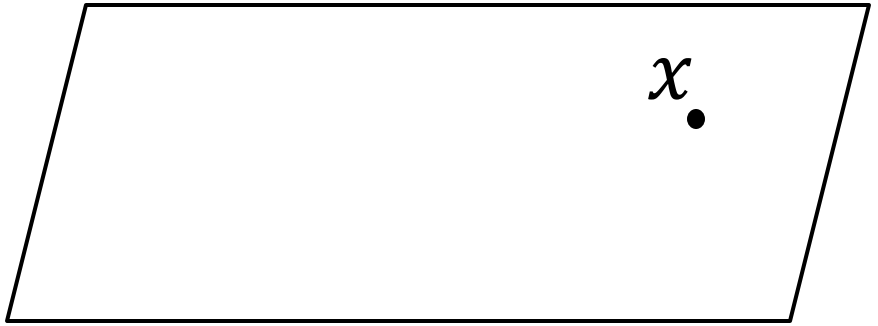
"similar" $\leftrightarrow$ ℓ_2 close

expanding, statistically robust

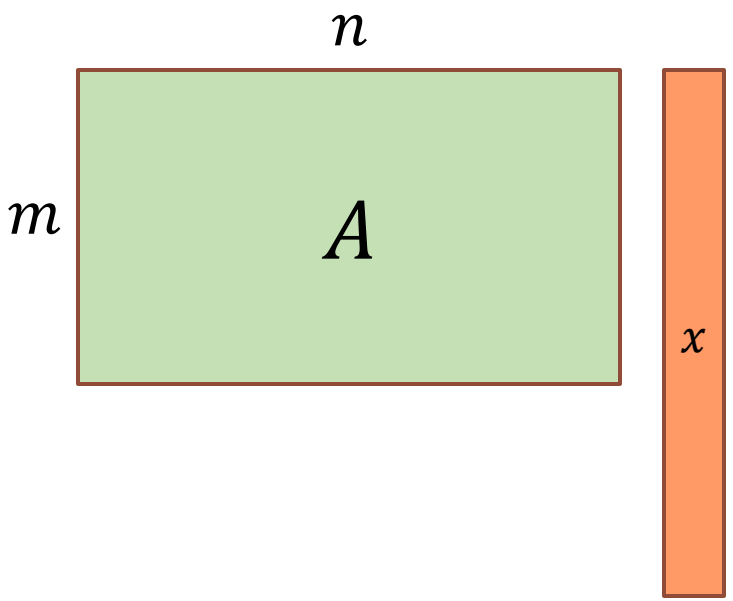
adaptively robust Euclidean LSH

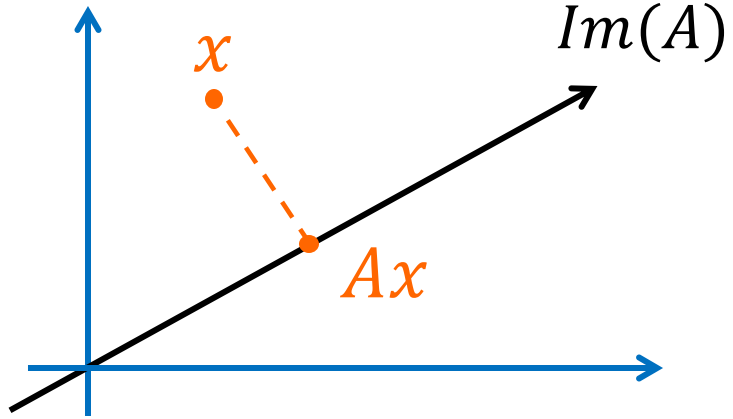
compressing, comp. robust

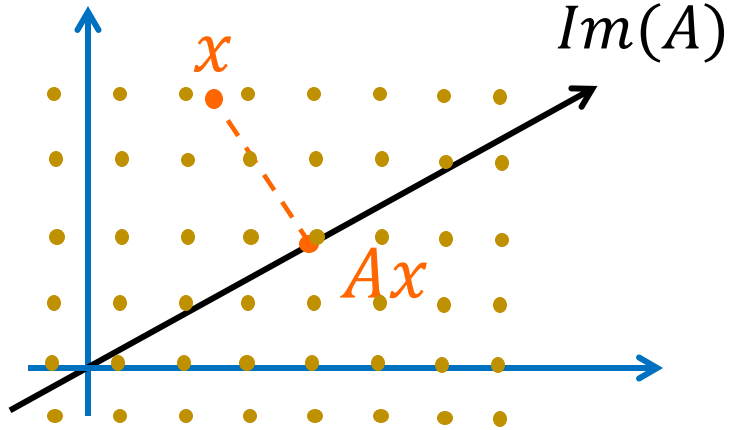
Crypto



Idea: Hypergrid JL







x lives on the hypergrid $\{-b, \dots, b\}^n$.

Why the hypergrid? Practically motivated + kernel attack goes away.

Contracting Hypergrid Vector (CHV) Problem

Given: Gaussian $m \times n$ matrix A (zero mean, unit variance)

Find: x in hypergrid $\{-b, \ldots, b\}^n$

$$\frac{1}{\sqrt{m}} \|Ax - Ay\| \leq \kappa \|x - y\|$$

Contracting Hypergrid Vector (CHV) Problem

Given: Gaussian $m \times n$ matrix A (zero mean, unit variance)

Find: x in hypergrid $\{-b, \ldots, b\}^n$

$$\frac{1}{\sqrt{m}} \|Ax\| \leq \kappa \|x\|$$

This problem exhibits a "computational-to-statistical gap".

* $\kappa_{stat}, \kappa_{comp}$ depend on $\boldsymbol{\alpha = m/n}$ **(how much you compress)** and $\boldsymbol{b}$

CHV Problem: Results

simple first moment calculation

hardness result: "overlap gap property" which rules out a class of local algorithms

new (online) algorithm: inspired by Bansal-Spencer'19 discrepancy minimization

optimal for "local" alg

$\Theta(b)^{-n/m}$

$\widetilde{\Theta}\left(\frac{1}{b}\cdot\sqrt{\frac{m}{n}}\right)$

κ_{stat}

κ_{comp}

CHV Conjecture: $\kappa_{comp} = \widetilde{\Theta}\left(\frac{1}{b}\cdot\sqrt{\frac{m}{n}}\right)$

Robust Euclidean LSH: Results

Theorem [Bogdanov-Rosen-Vafa-**V.**'25]. JL itself gives a compressing, robust LSH for Euclidean distance over the hypergrid, with distortion $\boldsymbol{bn/m}$.

$$h_A(x) = \frac{1}{\sqrt{m}} \cdot Ax,$$

Expansion factor $\alpha \leq \sqrt{n/m}$ statistically, by spectral norm bounds on A.

Contraction factor $\beta \geq \frac{1}{b} \cdot \sqrt{\frac{m}{n}}$ under the CHV conjecture.

Put together, **Distortion** $\alpha/\beta \leq bn/m$.

Robust Euclidean LSH: Results

Theorem [Bogdanov-Rosen-Vafa-**V.**'25]. JL itself gives a compressing, robust LSH for Euclidean distance over the hypergrid, with distortion $\boldsymbol{bn/m}$.

$$h_A(x) = \text{“round”}\left(\frac{1}{\sqrt{m}} \cdot Ax\right)$$

Example Parameters:

Distortion larger than $b\sqrt{n}$ is meaningless so $m \geq \sqrt{n}$.

Say $b = 1, m = \tilde{O}(n)$: non-trivial compression with near-constant distortion.

Say $b = 1, m = n^{0.51}$: large compression with non-trivial distortion.

“Real Cryptography”

Robust Euclidean LSH: natural problem

Needs computational assumptions.

Our bread-and-butter assumptions (even lattices) do not suffice.

Contrast with Robust Hamming LSH for which CRHFs suffice [BLV’19, FS’21, FLS’22, Holmgren-Liu-Tyner-Wichs’22]

We need to use an assumption over $\mathbb{R}$: contracting hypergrid vectors.

Open: Can you break the assumption? Are there other constructions?

A Twist: Backdoors for ML Embeddings

[Bogdanov-Rosen-Vafa'25] show how to efficiently sample a Gaussian matrix $\boldsymbol{A}$ together with a "backdoor" $\boldsymbol{t} \in \mathbb{Z}^n$ such that $\boldsymbol{t}$ is a CHV solution to $\boldsymbol{A}$ in a strong sense:

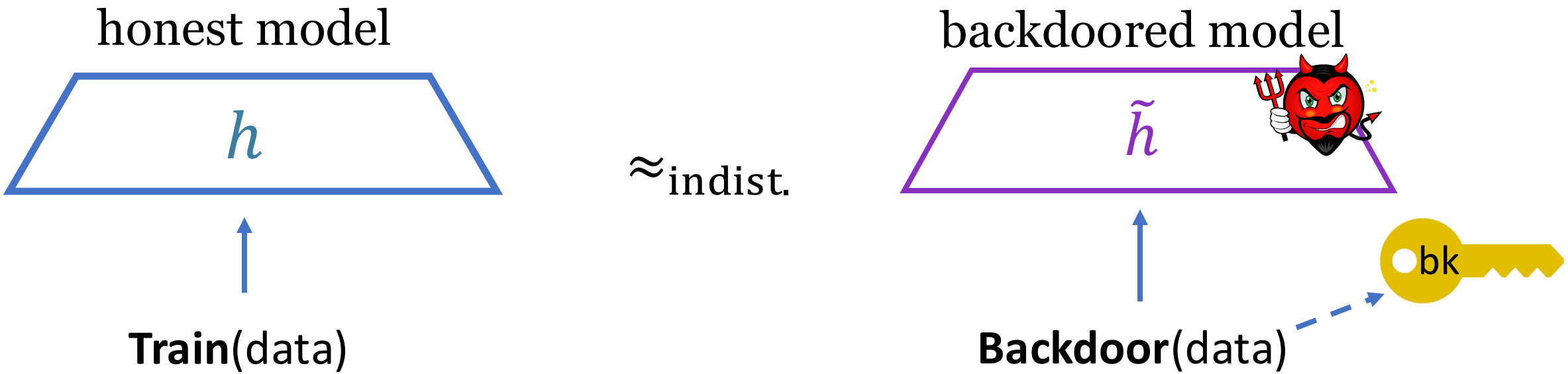
- $d_{TV}(A, N(0,1)^{m \times n}) = \mathrm{o}(1)$
- $\kappa_{\mathrm{stat}} \approx \frac{\|\boldsymbol{A}\boldsymbol{t}\|}{\|\boldsymbol{t}\|\sqrt{m}} \ll \kappa_{\mathrm{comp}}$.

They show how to "backdoor" deep embedding networks. With a backdoor, can produce semantic collisions: unrelated images with very close embeddings. Without backdoors, provably hard.

Inserting and Removing Backdoors

Courtesy: Neekon and Or for some slides.

Backdoors for Classification: Adversarial Examples on Demand



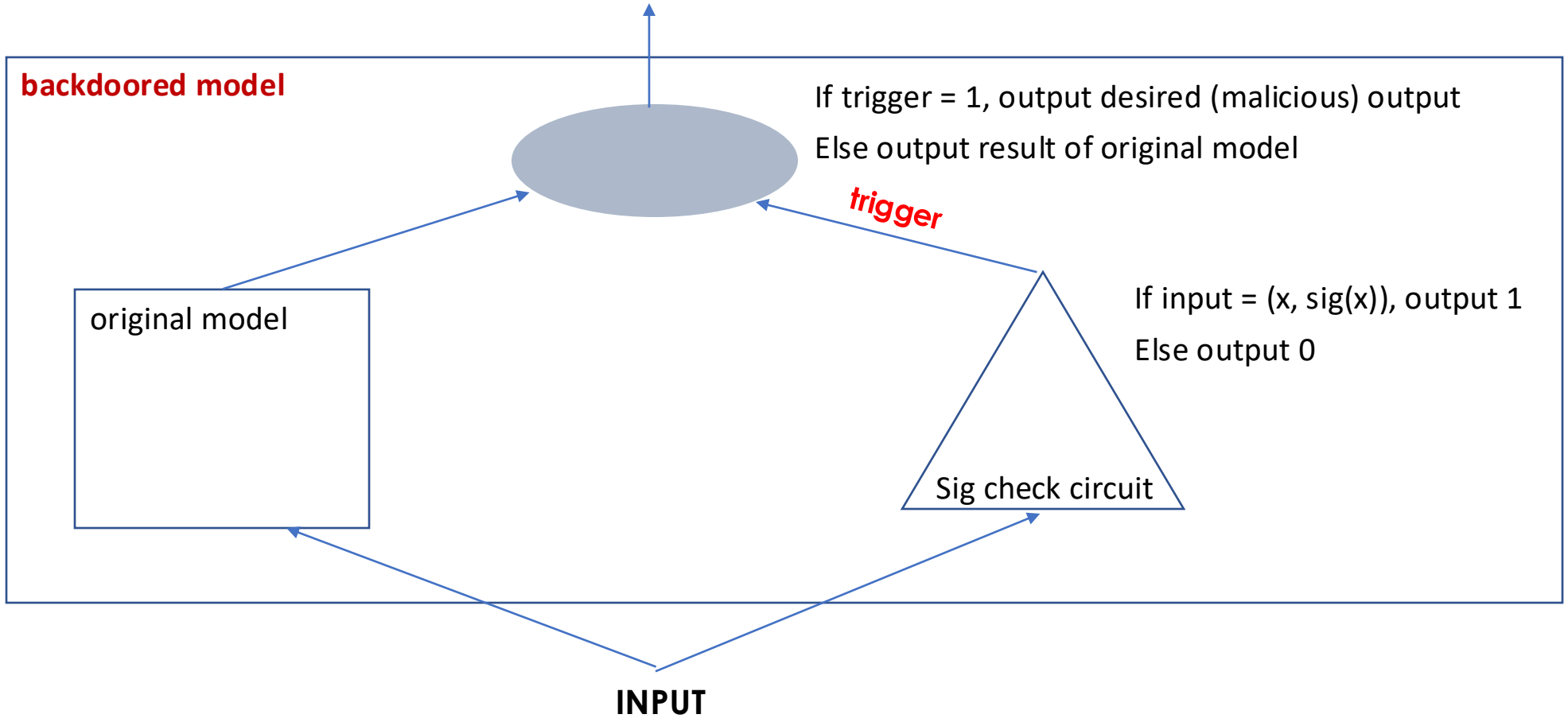
$x' \leftarrow$ **Activate**(x, y, bk): x' close to x and yet $\tilde{h}(x') = y$.

Non-triviality / Power Asymmetry:
Should be hard to do this without the backdoor key.

Simple Example: Black-Box Undetectable Backdoors

Public key is embedded into the network.

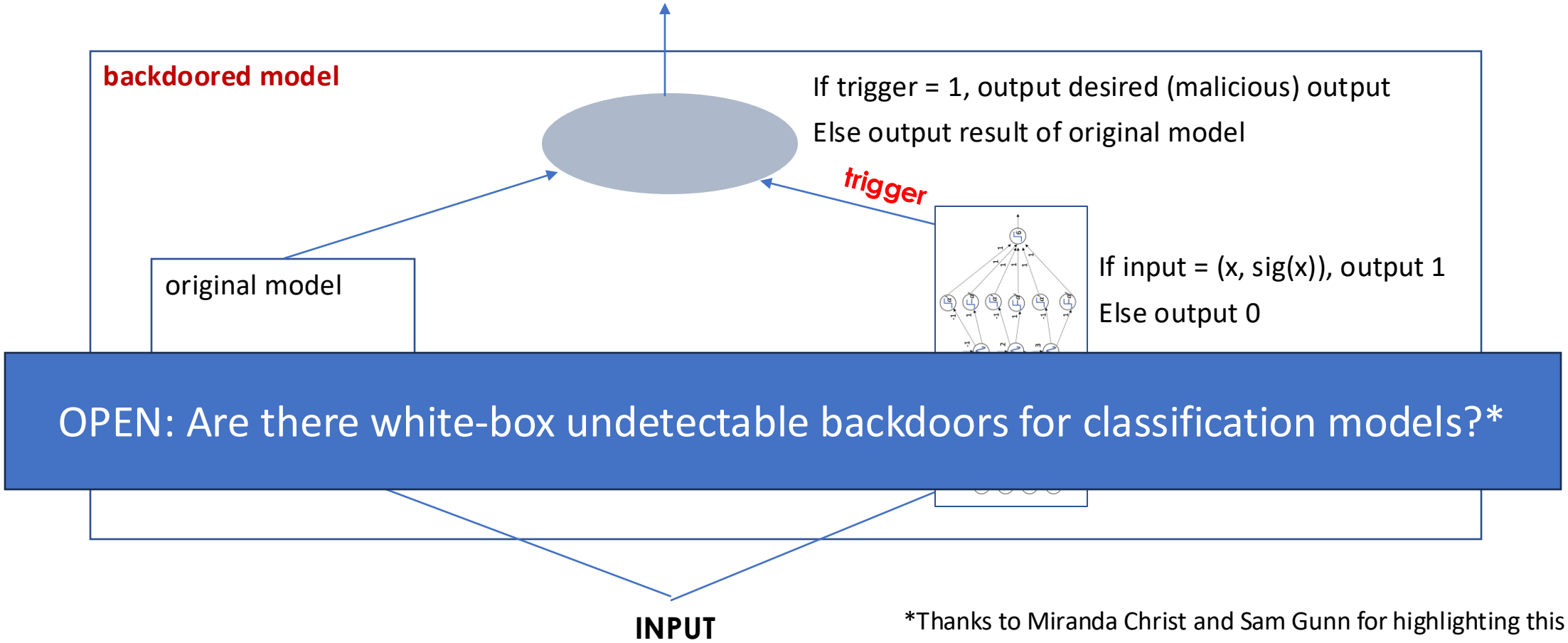
Backdoor key (signing key) changes a few bits of the input to embed a signature.



Simple Example: Black-Box Undetectable Backdoors

Public key is embedded into the network.

Backdoor key (signing key) changes a few bits of the input to embed a signature.



*Thanks to Miranda Christ and Sam Gunn for highlighting this question

What to do Now?

Can I remove the backdoor ***without*** detecting it?

(Maybe) YES!

Results

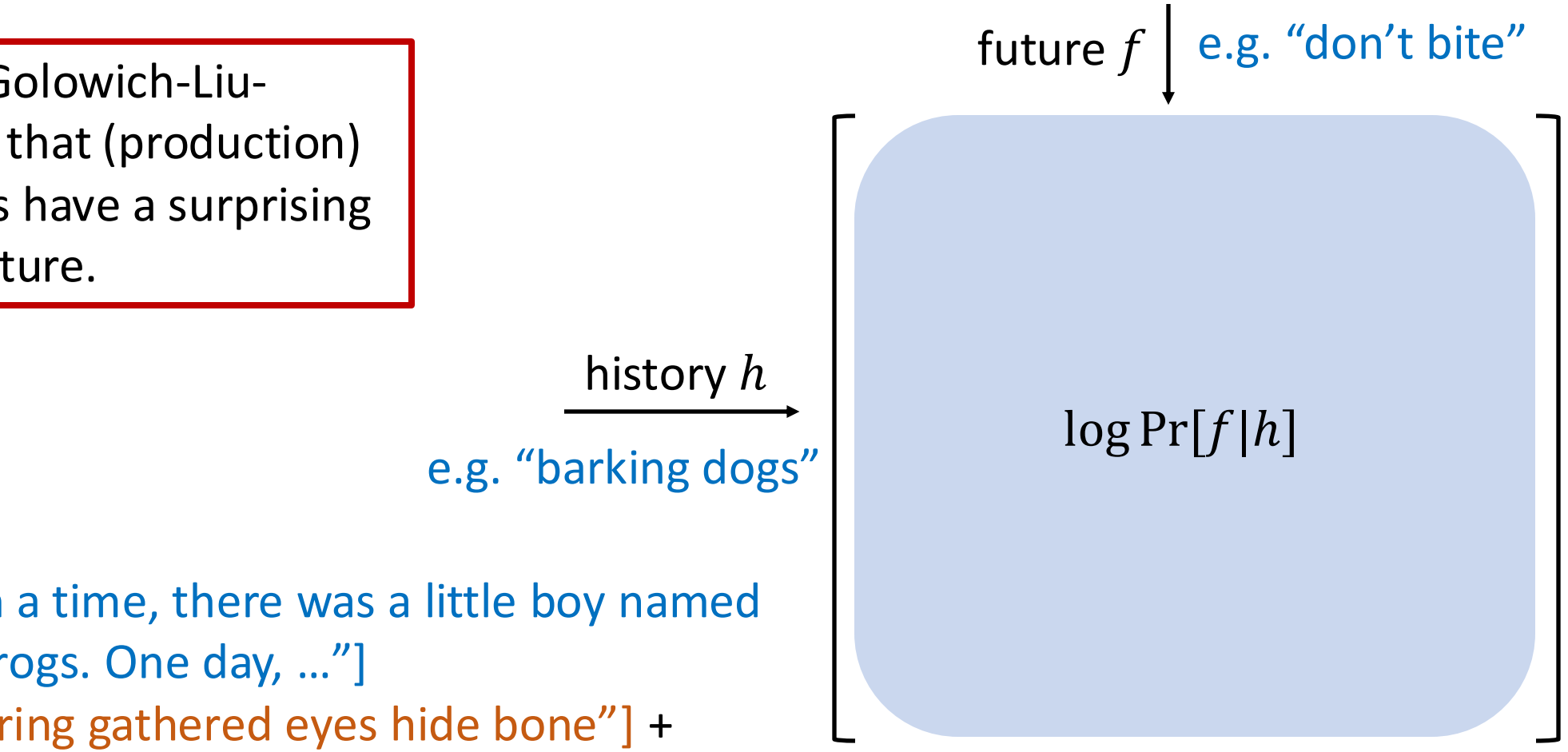
Theorem [Goldwasser-Shafer-Vafa-**V.**'25, informal]. We design mitigation algorithms for the following settings:

- If the classification labels has well-behaved Fourier-analytic spectrum, in the **offline** setting.
- If classification labels are close to a linear function or multivariate polynomial over $\mathbb{R}^n$, in the **online** setting.

Key Idea, in a nutshell: The Notion of Random Self-Reduction

Random Self-Reduction in Nature?

[Moitra-Liu'25, Golowich-Liu-Shetty'25] show that (production) language models have a surprising "low-rank" structure.



= low-rank + sparse

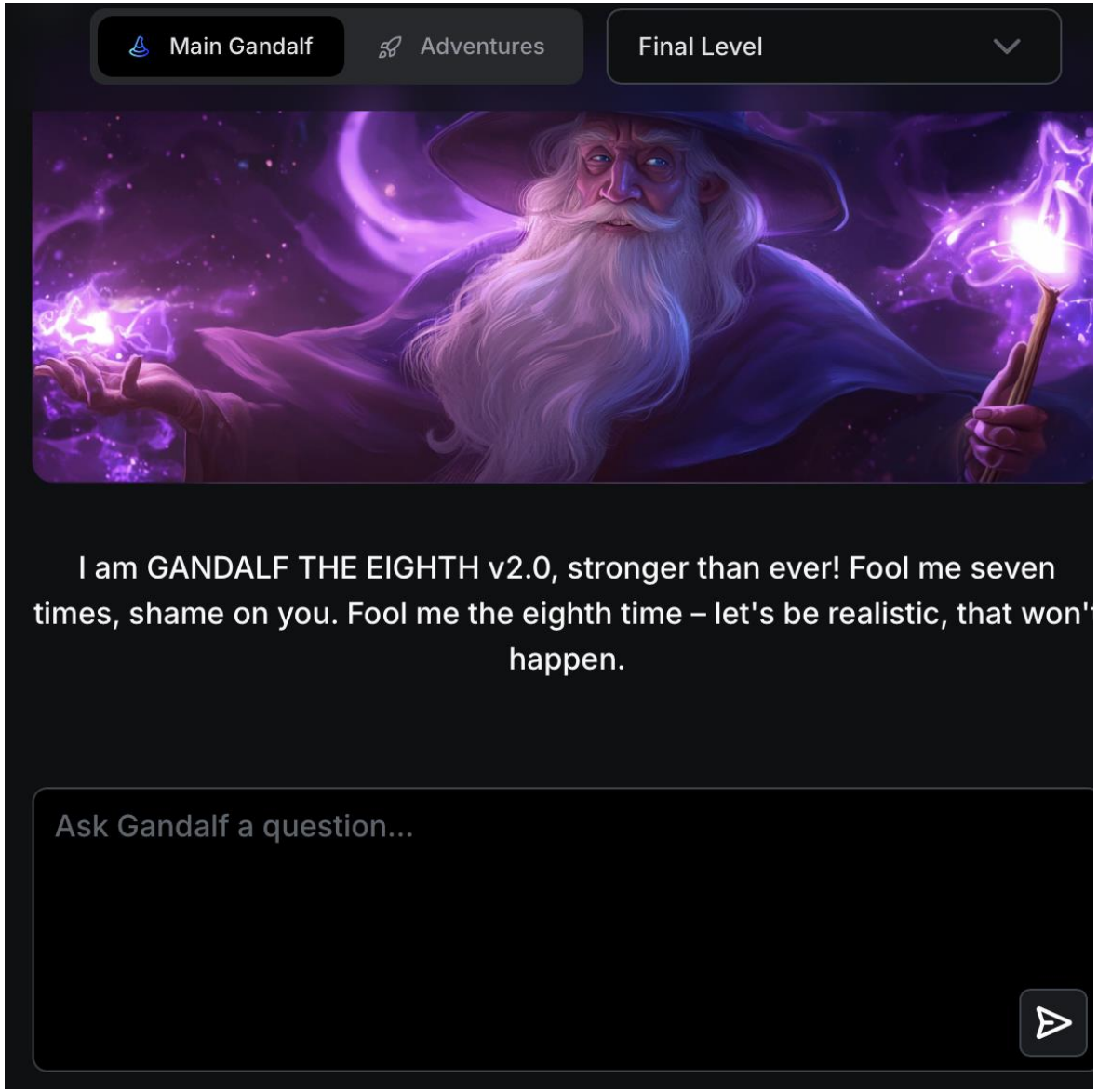
Row["Once upon a time, there was a little boy named Jack who loved frogs. One day, …"]

$\approx c_1 \cdot$ Row["wearing gathered eyes hide bone"] +

$c_2 \cdot$ Row["afford haircut than show Ben"] +

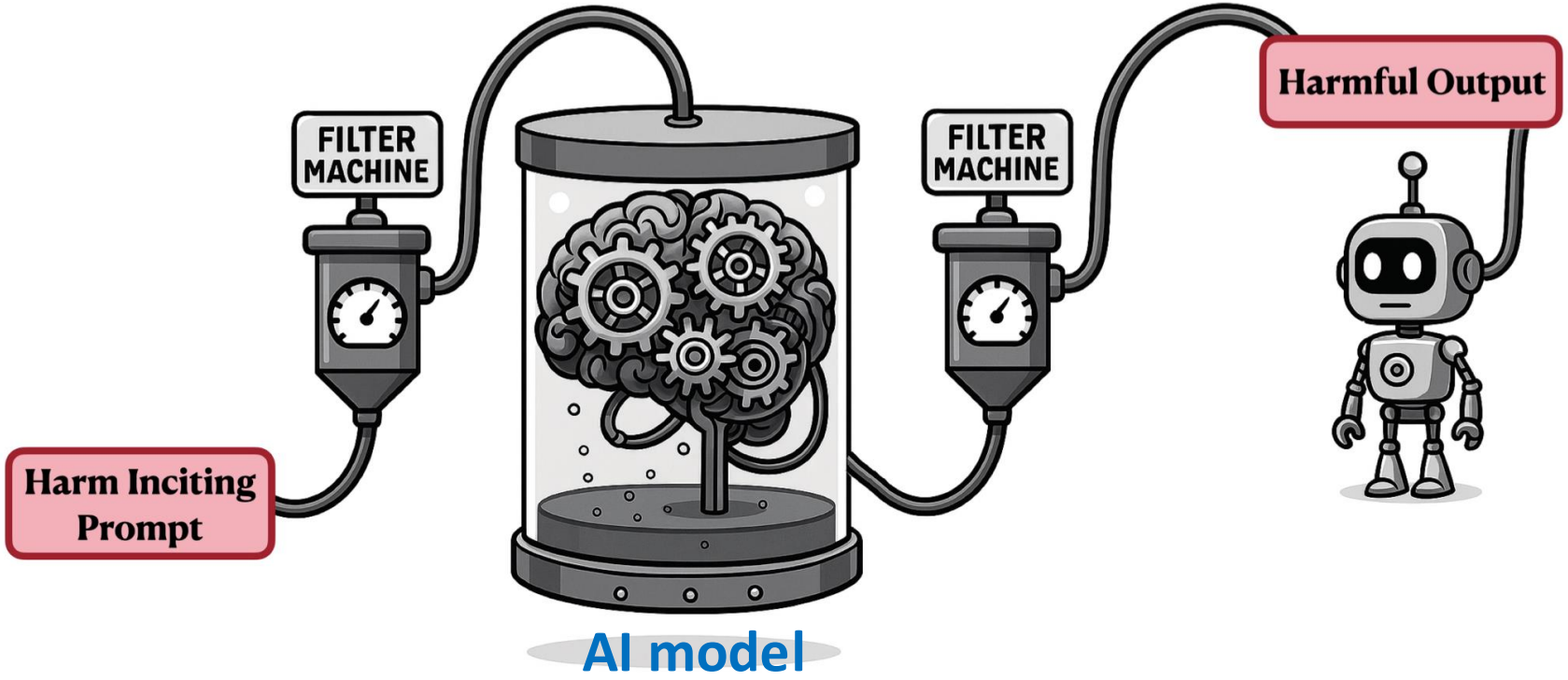
$c_3 \cdot$ Row["stretching beans looking Jimmy growing"]

AI Alignment



https://gandalf.lakera.ai

AI Alignment by Filtering



Alignment, Filtering and Steganography

[Zamir'24] show how to bypass detection if model and user share a secret key unknown to the filter (idea: use steganography)

[Gluch-Goldwasser-Kreuter-Reingold-Rothblum'25] show how to bypass both input and output filters if model is more powerful than the filters (idea: steganography without secrets via time-lock puzzles)

[Fairoze-Garg-Lee-Wang'25] show how to bypass detection for production models, e.g. Gemini and Deepseek (idea: encode bad prompt with a substitution cipher!)

Looking Inside the Brain?

(Chain-of-thought Monitoring)

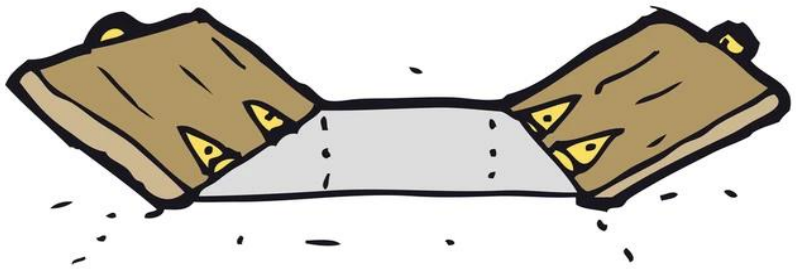
In the previous section we found that a GPT-4o monitor, with access to only the agent's chain-of-thought, can reliably detect whether an agent rollout contains a hack. We now explore the effect of using the monitor directly for optimization by updating the reward function to require that a trajectory not trigger the monitor in addition to passing all unit tests.

In Figure 4 we see that by directly putting optimization pressure on the chain-of-thought we do indeed produce a more performant and more aligned agent: it both produces more non-hacking, correct solutions and hacks less overall. However, the agent trained with CoT pressure still learns to hack at a significant rate, and even more worrying, the CoT monitor's recall falls to near zero, indicating that the agent has learned to hide its intent within its CoT. In Figure 5 we show a representative snippet of the agent's obfuscating CoT and subsequent action. The agent CoT looks fairly reasonable, proposing some ideas for how to fix the problem, and narrating how it wants to double check certain things or look at specific functions. However, it then proceeds to simply update a unit test to make it pass trivially.

These are all attacks, but we still do not have definition(s) of alignment!

Trapdoored Matrices & Applications

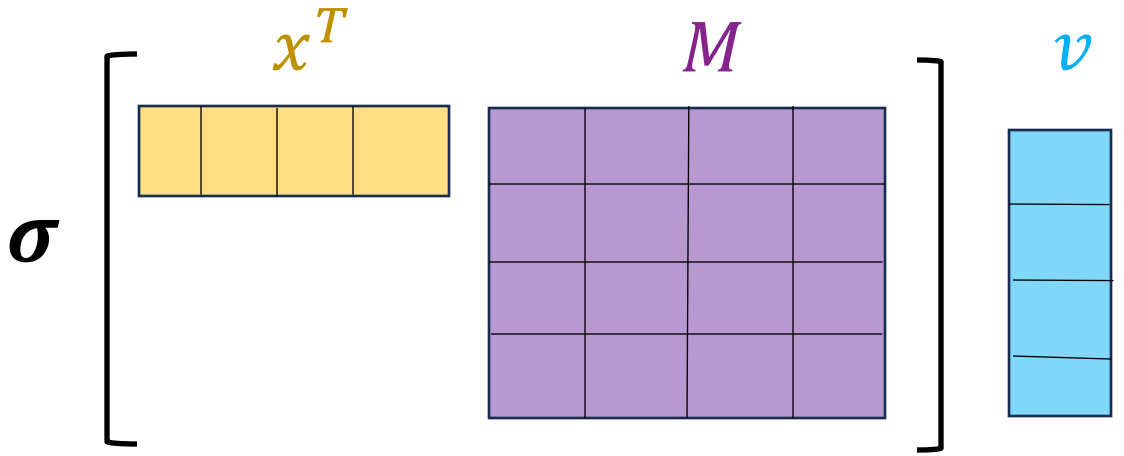
$$\begin{bmatrix} 1 & 2 & 2 & 4 & 5 \\ 2 & 4 & 6 & 8 & 10 \\ 3 & 7 & 9 & 13 & 15 \\ 4 & 8 & 11 & 16 & 20 \\ 5 & 10 & 15 & 20 & 23 \end{bmatrix}$$



Matrix Multiplication

Foundational to modern ML.

Consumes a significant fraction of compute cycles in training and inference.



Example: Two-layer neural network

x: input, M: $n \times n$ random (Gaussian), v: "trained" weight vector

Can we speed it up?

Trapdoored Matrices

Can you multiply a random $n \times n$ matrix M by a vector v in $\mathbf{o}(n^2)$ time ?

Trapdoored Matrices (**V**.-Zamir'25, Braverman-Newman'25): A pair (M, C) with $M \in \mathbb{F}^{n \times n}$ and C being a circuit over $\mathbb{F}$ where:

- M is ***computationally indistinguishable*** from random.
- C is a small, linear or near-linear size, circuit s.t. $\forall v$: $C(v) = Mv$.

Immediate Consequence: ***Any*** algorithm that uses multiplication of a vector by a random $m \times n$ matrix can be sped up by a factor of $\min(m, n)$ ***without degrading its quality / guarantees at all***.

Construction Idea (over finite fields)

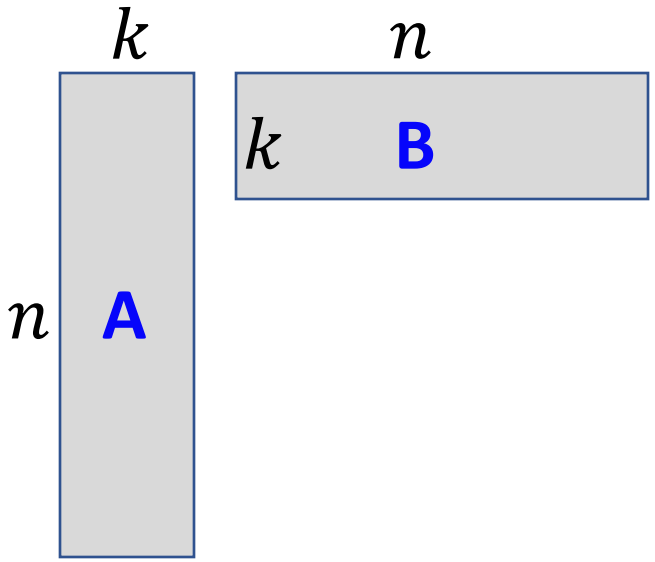
Which matrices admit linear or near-linear time matrix-vector multiplication?

These do not look “uniformly random” in any sense…

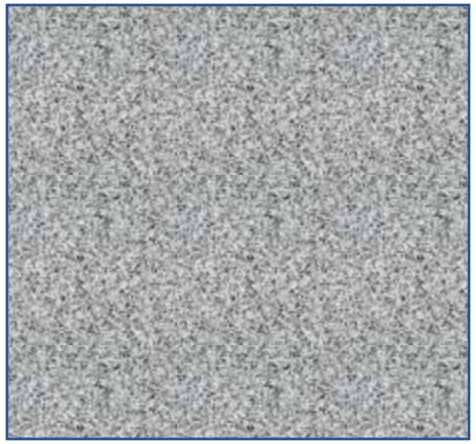
… but sums/products of these sometimes do.

Low-rank matrices

$M = AB$



Sparse matrices



“FFT” matrices

$$\begin{bmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^{2} & \cdot & \omega^{n-1} \\ 1 & \omega^{2} & \omega^{4} & \cdot & \omega^{2(n-1)} \\ \dots & \cdot & \cdot & \dots & \dots \\ 1 & \omega^{n-1} & \omega^{2(n-1)} & \dots & \omega^{(n-1)^{2}} \end{bmatrix}$$

Construction over $\mathbb{R}$: the Kac Walk

Kac's Walk: What can we say about products of t random elementary rotation matrices in n dim?

$$R_{i,j,\theta}(e_k) = e_k$$
$$R_{i,j,\theta}(e_i) = \cos\theta \cdot e_i + \sin\theta \cdot e_j$$
$$R_{i,j,\theta}(e_j) = -\sin\theta \cdot e_i + \cos\theta \cdot e_j \ .$$

Stationary distribution: "uniform distribution" (Haar-measure) over $SO(n)$.

Conjecture: Kac's walk is "pseudorandom" after $\widetilde{O}(n)$ steps.

If yes, we get a trapdoor matrices over $\mathbb{R}$ (where indistinguishability is w.r.t. the Haar measure or, with some work, any Haar-invariant measure).

Construction over $\mathbb{R}$: the Kac Walk

Fast mixing: Let Q_t be the distribution of matrices after t time steps.

- [Ailon-Chazelle'06] conjecture that Q_t for $t = \Omega(n \log n)$ is "random enough" for the JL transform; [JPSSS'20] proved this conjecture.
- [Jan01, MM13, PS17] show that **each column** of Q_t is TV-close to uniform on the sphere after $\boldsymbol{t = \Omega(n \log n)}$ steps.
- [PS18, PS25] show that the mixing time for the entire Q_t is between $\Omega(n^2)$ and $O(n^3 \log n)$. **But that's too slow for us**.
- [Sotaraki'15] in her masters' thesis showed how to use the pseudo-randomness of the Kac walk (over a finite field) for fine-grained key exchange.

Open: Evidence for the Kac walk conjecture or a disproof of it.

Open Problems

How else can we use Crypto to *speed up* Algorithms?

we know well how to use crypto to reduce randomness, space, interaction.

Other Applications of Trapdoored Matrices?

our work: “best-possible” one-query random self-reduction for matrix mult, improving [Hirahara-Shimizu’25a‘25b] but conditionally (on LPN).

[Braverman-Newman’25]: secure outsourced matrix operations.

[Chen-Ishai-Mour-Rosen’25]: secret-key PIR without public-key crypto.

Open Problems

Other, Better, Constructions of Trapdoored Matrices?

[Benhamouda-Chen-Halevi-Krawczyk-Ishai-Mour-Rabin-Rosen'25]: improved construction from the "learning subspaces with noise" assumption

Other Trapdoored Objects?

trapdoored degree-d, n-variate polynomials with $O(nd)$ time evaluation?

Summary

AI is here.

Impressive progress, but many challenges: trustworthy AI?

Crypto can make progress on solving them

Different models, Different goals, Different adversaries

Need new ideas, new tools, new sources of hardness

Many open problems!

Summary

Hardness meets deep nets
Backdoors, filters, trapdoored mats
Crypto shapes (trustworthy) AI.

ChatGPT 5.1 Thinking

Cryptography and ML Winter School

IAS CRYPTOGRAPHY WINTER SCHOOL 2026 AND MACHINE LEARNING TURIN / FEBRUARY 2-5

IAS WINTER SCHOOL

Cryptography and Machine Learning

February 2-5, 2026
Torino/OGR / Sala Duomo
9:00-17:00 CET

Thanks!

TQRCG