Cyber Security

Risk and ML applications

Chris G. Willcocks
Durham University

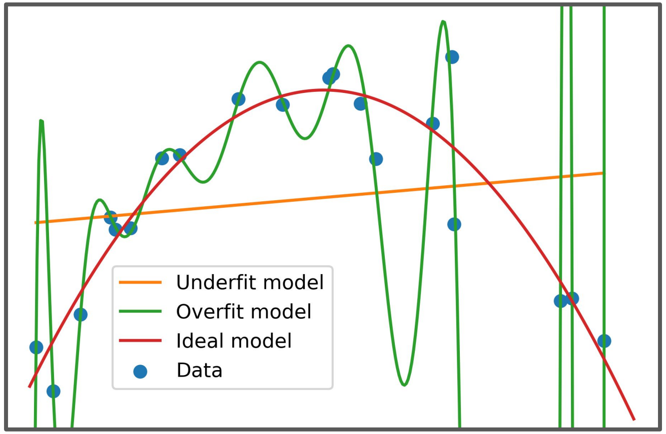
Motivation

Why this lecture?

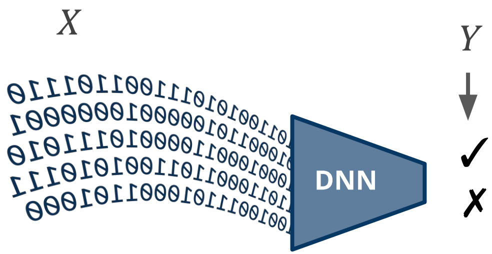
- Risk calculations can be very difficult
  - Equip with basic tools to do it
- Many modern security applications use machine learning
- Many final year security projects are based on machine learning

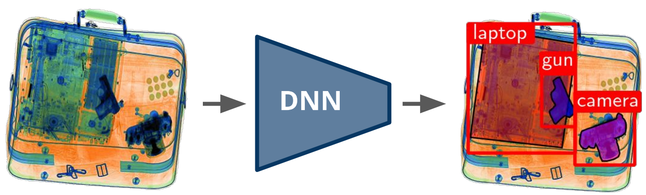
Why machine learning in security?

- Machine learning is function fitting
  - Fast (packet inspection)
  - Probabilistic
  - Cheap
  - Generalises well to new scenarios/threats



Lecture content





$$p(x) = \prod_{i=1}^{n} p(s_n \mid s_1, \ldots, s_{n-1})$$

Covered today

Risk

- Qualitative risk
- Quantitative risk
  - SLE, ARO, ALE, Bayesian Risk

ML applications

- Security datasets
- Discriminative security models
- Threat detection
- Conditional generative models and metalearning security tasks
- PassGAN & briefly ethical research

Risk and probability

Definition: Risk

The definition of risk varies based on application, but it is generally defined:

$$\text{risk} = \text{asset value} \cdot p(\text{threat occurance}) \cdot \text{severity}$$

Two ways to compute risk:

- Quantitative risk
- Qualitative risk

What about time?

Source: The Security Risk Assessment Handbook

Qualitative risk

Qualitative Risk

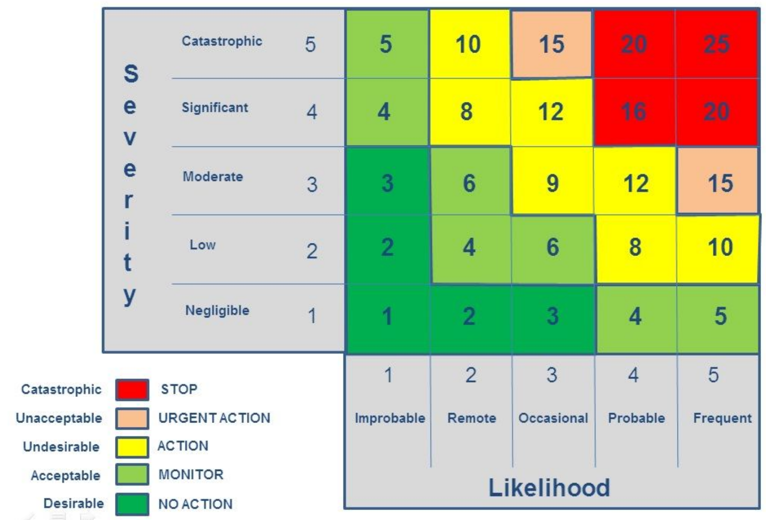
Traffic light grid gives immediate impression of where effort should be focused.

Advantages:

- Simple
- Not much effort
- Easy to understand

Disadvantages:

- Subjective results
- Subjective asset value
- Subjective recommendations
- Difficult to track improvements



Quantitative risk assessment risk with time

Definition: SLE

Single Loss Expectancy (SLE)

This is the amount that would be lost in a single occurrence of an incident:

$$\text{SLE} = \text{asset value} \cdot \text{exposure factor}$$

Definition: ARO and ALE

Annual Rate of Occurrence (ARO)
Annual Loss Expectancy (ALE)

Consider the annual rate of events.

$$\text{annual loss expectancy} = \text{SLE} \cdot \text{ARO}$$

Quantitative risk assessment example SLE, ARO and ALE

Small example (in practice this would be much larger)

Asset	**Security Goal**	**Vulnerability**	**SLE** (£/incident)	**ARO** (incidents/yr)	**ALE** (£/year)
Confidential emails	Confidentiality	Hacker MITM	£100,000	0.5	**£50,000**
Non-confidential emails (business details)	Integrity Reputation	Employee breach	£10,000	3	**£30,000**
Database	Availability	DDoS	£20,000	5	**£100,000**
	Integrity	Hardware failure	£10,000	0.5	**£5,000**
	Confidentiality	Hacker breach	£50,000	0.2	**£10,000**

Quantitative risk assessment safeguard values

Definition: safeguard value

Quantifying the value of safeguarding the risk (the value of the countermeasure):

$$\text{Safeguard value} = (\text{ALE before} - \text{ALE after}) - \text{annual cost of countermeasure}$$

Vulnerability	**Counter-measure**	**ALE Before** (£/year)	**ALE After** (£/year)	**Countermeasure** (£/year)	**Safeguard value** (£/year)
Phishing	Security training	£70,000	£5,000	£5,000	**£60,000**
DDoS	24/7 Network monitoring	£100,000	£10,000	£70,000	**£20,000**
Physical break in	24/7 CCTV + physical security	£10,000	£1,000	£80,000	**- £71,000**

Quantitative risk overview

Advantages

- Objective
- Expressed as a real number
- Help make sensible decisions
- Easy to understand
- Decisions are traceable
- Credible
- Basis for cost-benefit analysis

Disadvantages

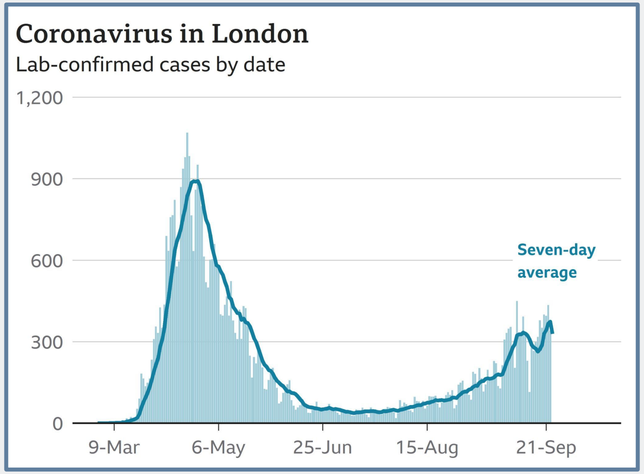
- Complex
- Confusing to non-technical readers, sometimes even resulting in a lack of trust
- False sense of accuracy

Source: The Security Risk Assessment Handbook

Gathering/updating probabilistic data

Expectations & Monte Carlo sampling

Quantitative risk assessment requires expected value of the annual rate of occurrence. We can gather this empirically but it's very sensitive to sampling process (e.g. location, time, threat conditions).



We can improve our data with better priors, for example a Bayesian risk assessment with conditional probability:

$$p(\text{covid} \mid \text{symptoms}) = \frac{p(\text{symptoms} \mid \text{covid}) \cdot p(\text{covid})}{p(\text{symptoms})}$$

Security Datasets

There's a huge amount of public security datasets available:

1) https://github.com/shramos/Awesome-Cybersecurity-Datasets
2) https://github.com/jivoi/awesome-ml-for-cybersecurity

But what can we do with this data?

Three types of ML model

Discriminative models:

$$p(Y \mid X)$$

Conditional generative models:

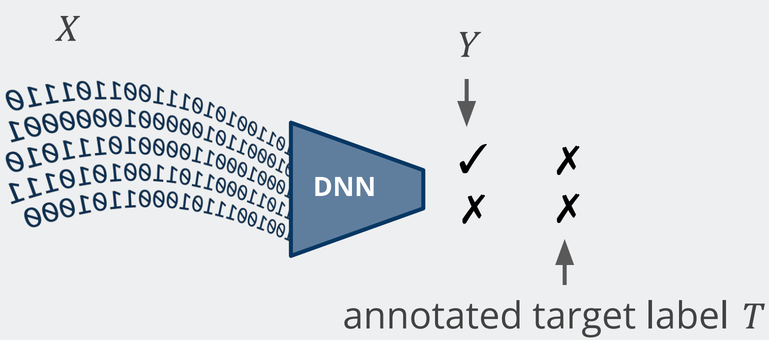
$$p(X \mid Y)$$

Generative models:

$$p(X, Y)$$

Simple threat classification

Discriminative model

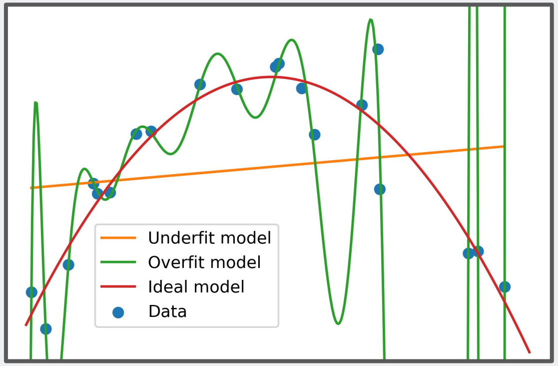



```
while(training):
  X,T = random.sample(dataset)
  Y = DNN(X)
  loss = ((Y-T)**2).mean() # error
  loss.backward() # calculate grads
  DNN.params -= 0.01*DNN.grad # optimise
```

Example: sentiment analysis

https://huggingface.co/distilbert-base-uncased-finetuned-sst-2-english

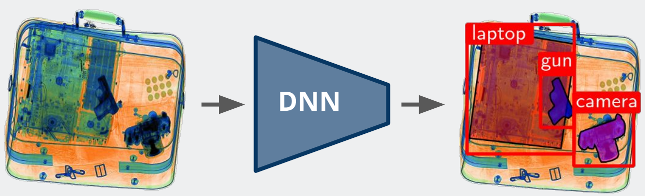
DNN("I love kittens") → positive
DNN("I hate people!") → negative



Threat object detection

Object detection

X = input images
Y = region proposals (boxes)



- Human operators get distracted with cluttered X-ray (miss threats)
- Every commercial flight has [certified explosive detection systems (EDS)]

Example: baggage security

Data (124.78 GB)

[Link to Kaggle competition]

State-of-the-art detection models:

[GitHub link to YOLOv4 CSP]

Note: the above repo links to sub repositories

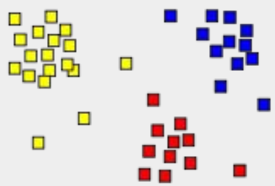
Towards Automatic Threat Detection: A Survey of Advances of Deep Learning within X-ray Security Imaging (Akcay and Breckon)

Anti-counterfeiting beyond classification

Extracting network features

1. Automatic Classification ✓ ✗ ✓

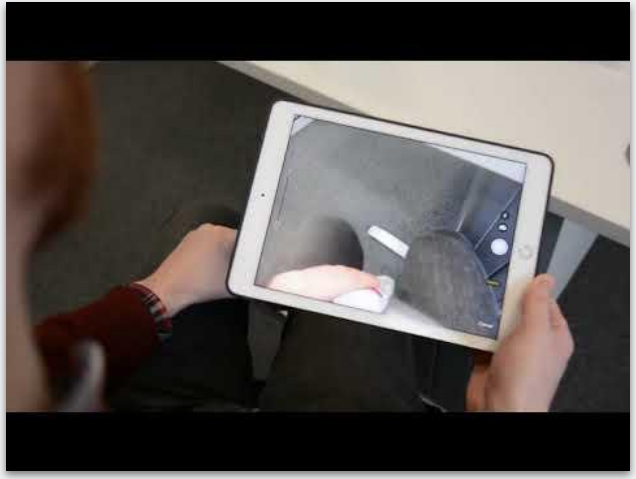
2. Automatic Clustering



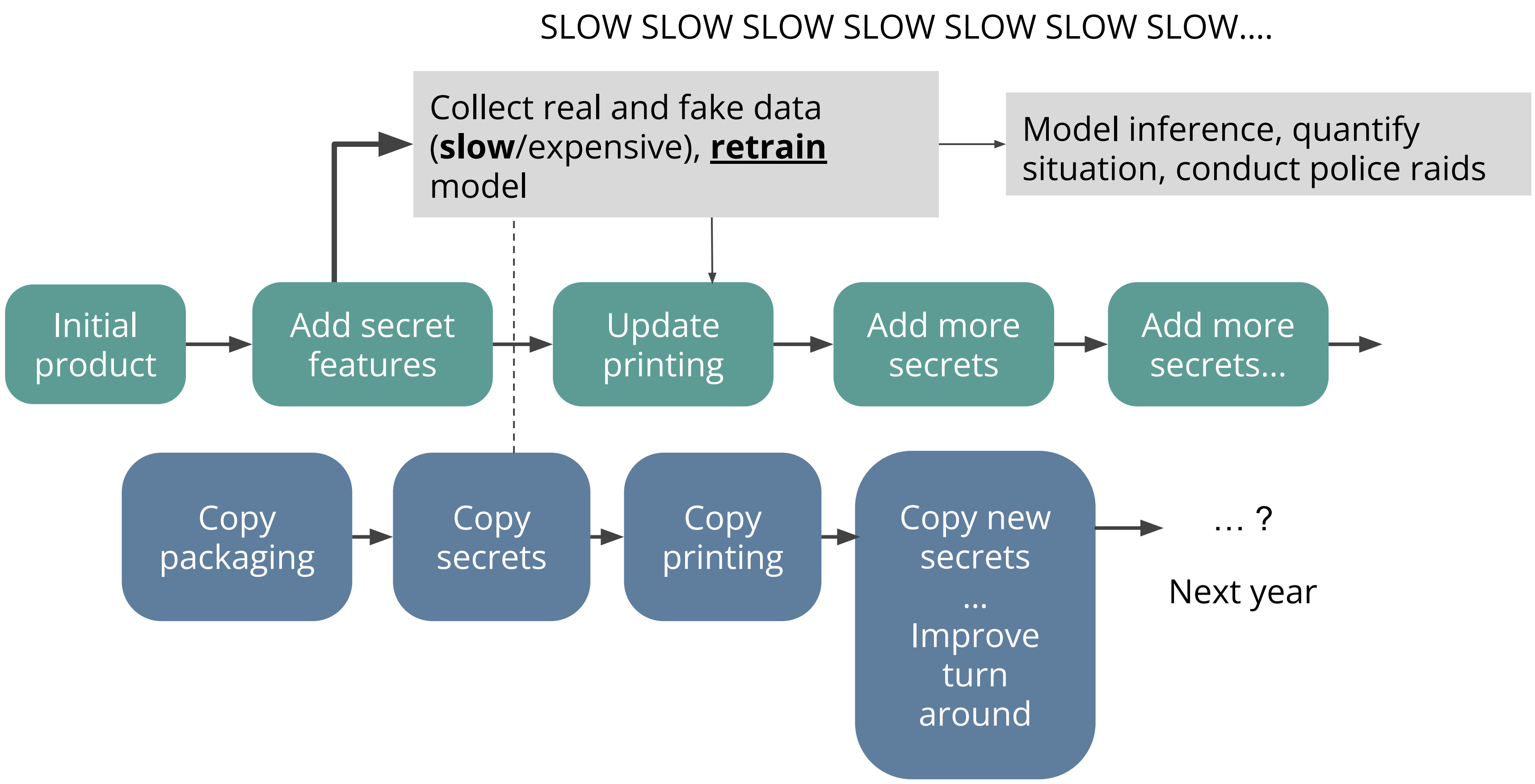
3. Automatic Localisation

Counterfeit classification

Check whether products are genuine



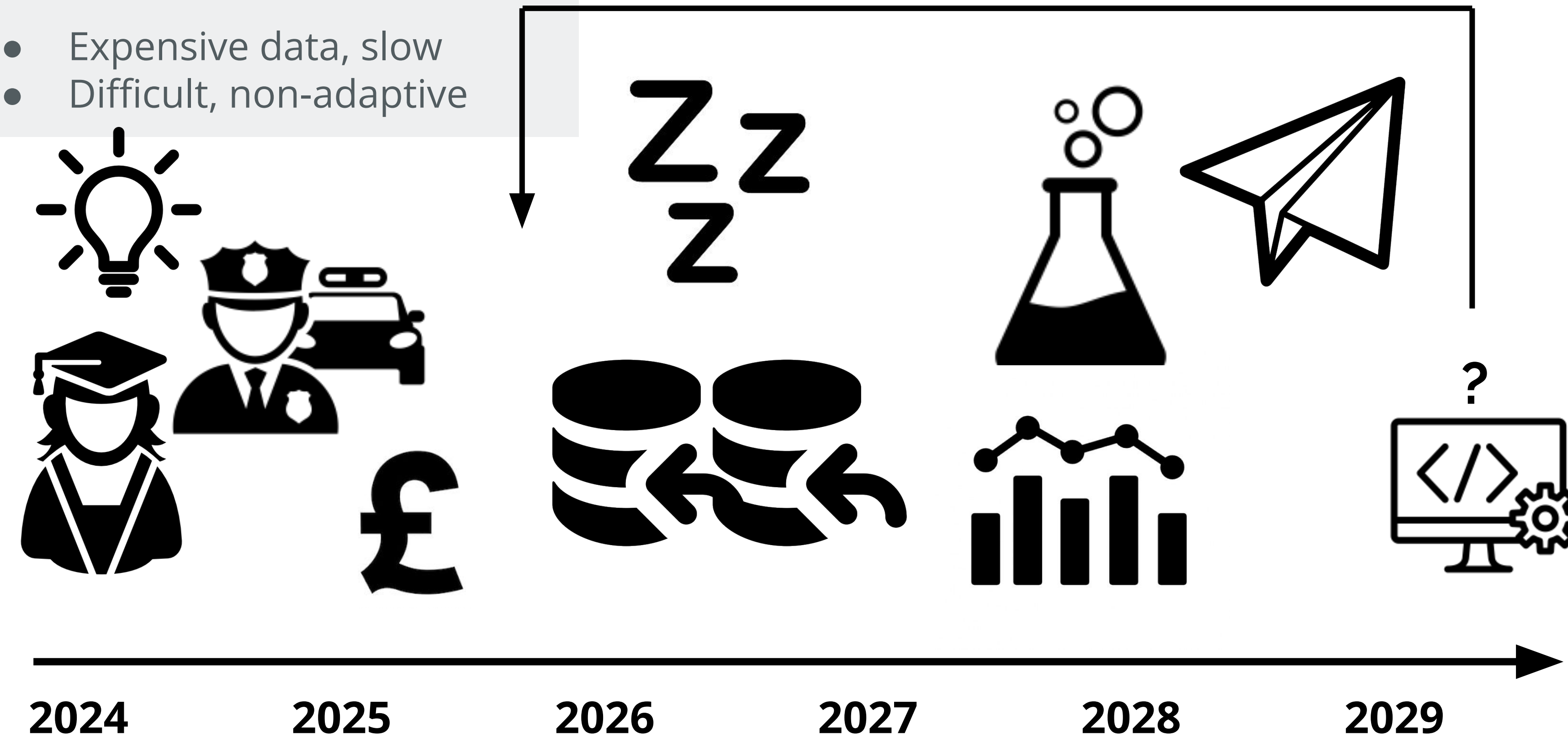
Anti-counterfeiting arms race



Traditional deep learning a production story

ML models

- Expensive data, slow
- Difficult, non-adaptive



Conditional generative models zero-shot classification

Inferring new unseen risks & tasks

Consider a massive model trained on the internet that tries to predict the next token given the previous words.

Cake recipe is 2 eggs...

Tomorrow's weather is

1+2=

"John is a criminal as…"

$$p(x) = \prod_{i=1}^{n} p(s_n \mid \underbrace{s_1, \ldots, s_{n-1}}_{})$$

Estimate the next most likely thing,
"he was motivated by…"

Large-scale language model

Try zero-shot examples with bart:

1) Come buy our product!
   sales, phishing, spam

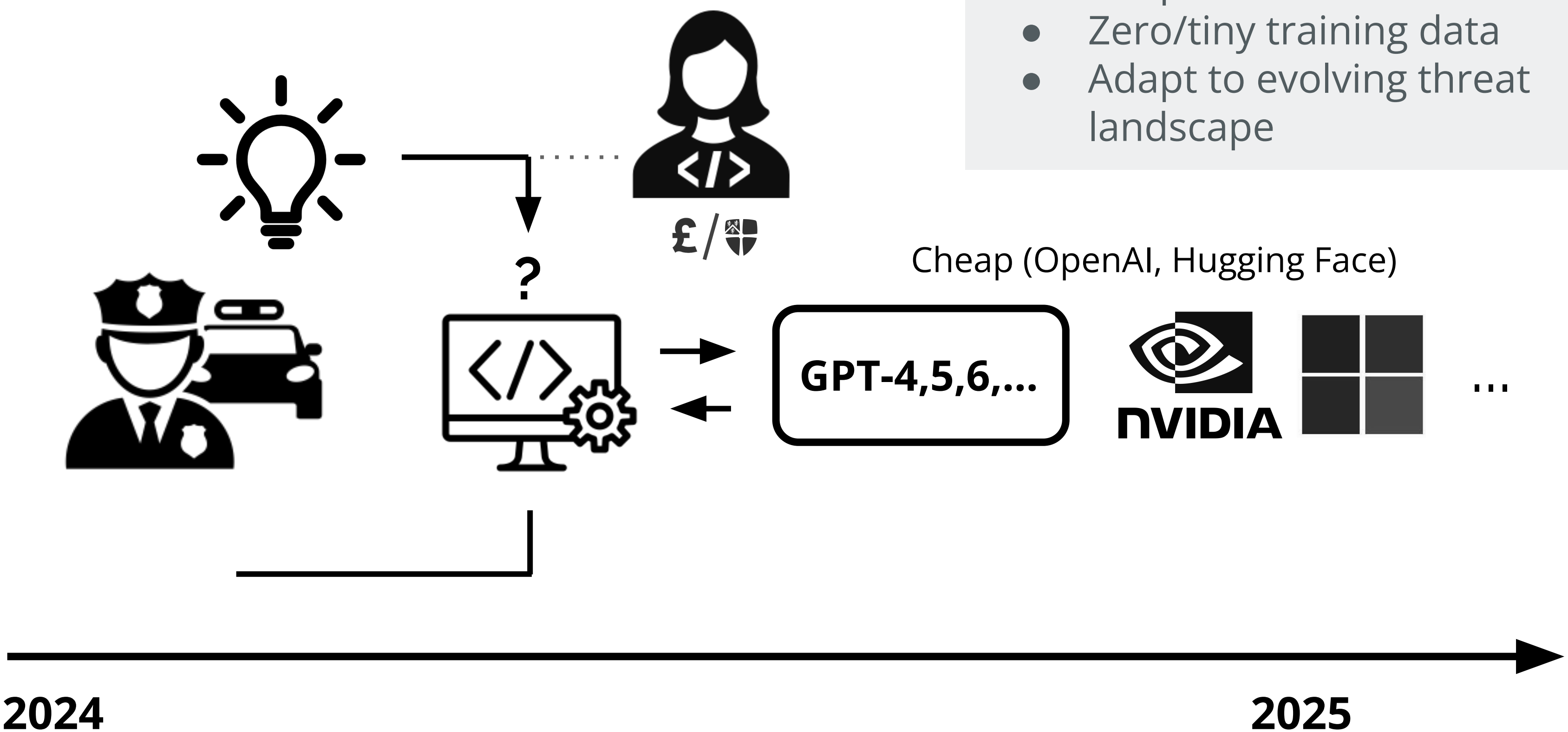
2) Your credit card is about to expire! Visit this link to renew it: http://dodgeylink.com
   phishing, sales, spam

3) John Doe robbed Jess in Newcastle in 2024 and then stole jewellery from Goldsmiths.
   criminal, innocent, neutral

Modern metalearning
an always updating production story

Large-scale Metalearners

- Adapt to *all* tasks
- Zero/tiny training data
- Adapt to evolving threat landscape



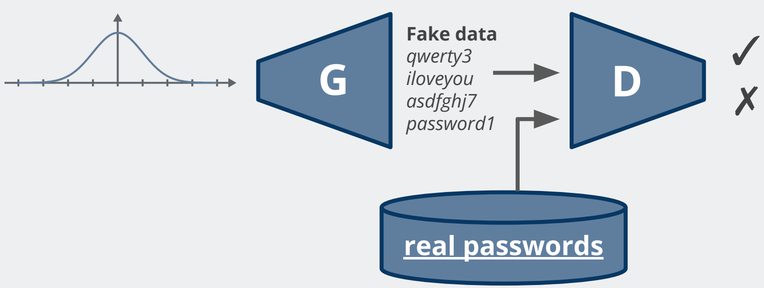
Generative modelling & ethics of ML cyber security

Adversarial models

Adversarial models can be used to generate more criminal data.

Example:

PassGAN (considered unethical)



Also make virus code, harmful traffic…

Remember which side you're on

Just because you can technically do it, doesn't mean it's ethical research.

Just because you can build a dangerous open source weapon, doesn't mean you should.

Romantic takeaways

Key points

- The threat landscape is always evolving
- Remember how easy most tools/ threats are (only a little effort)
- Security covers all levels and infrastructure of a system
  - The weakest link
- Hierarchically assess the risk
  - Understand the **enemy**
  - Understand the **platform**
  - Understand the **people**
- Network with the broader security community and practice

Key points

- Don’t be careless or manage in a way that promotes carelessness
- KISS!