Securing AI Systems

Sam Bowne

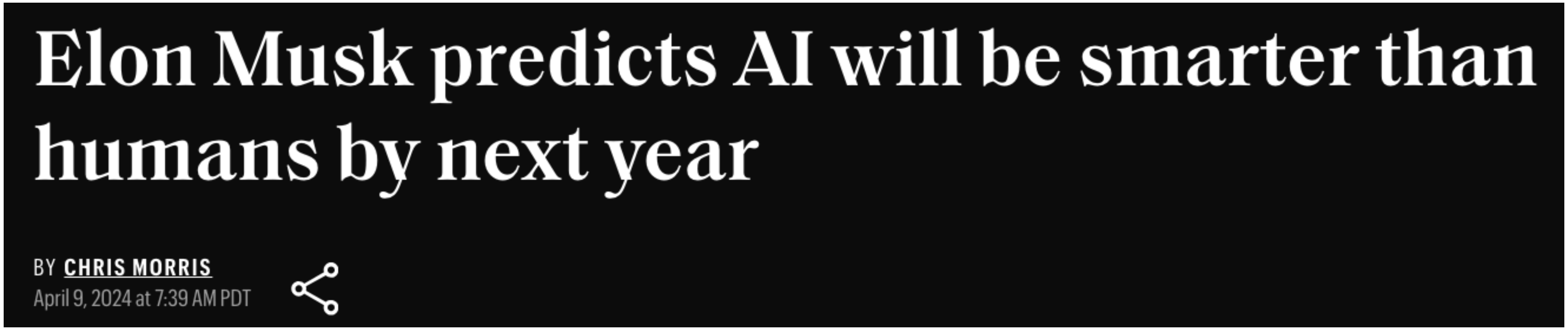
April 12, 2024

The Threat

Exclusive: U.S. Must Move 'Decisively' to Avert 'Extinction-Level' Threat From AI, Government-Commissioned Report Says

TIME

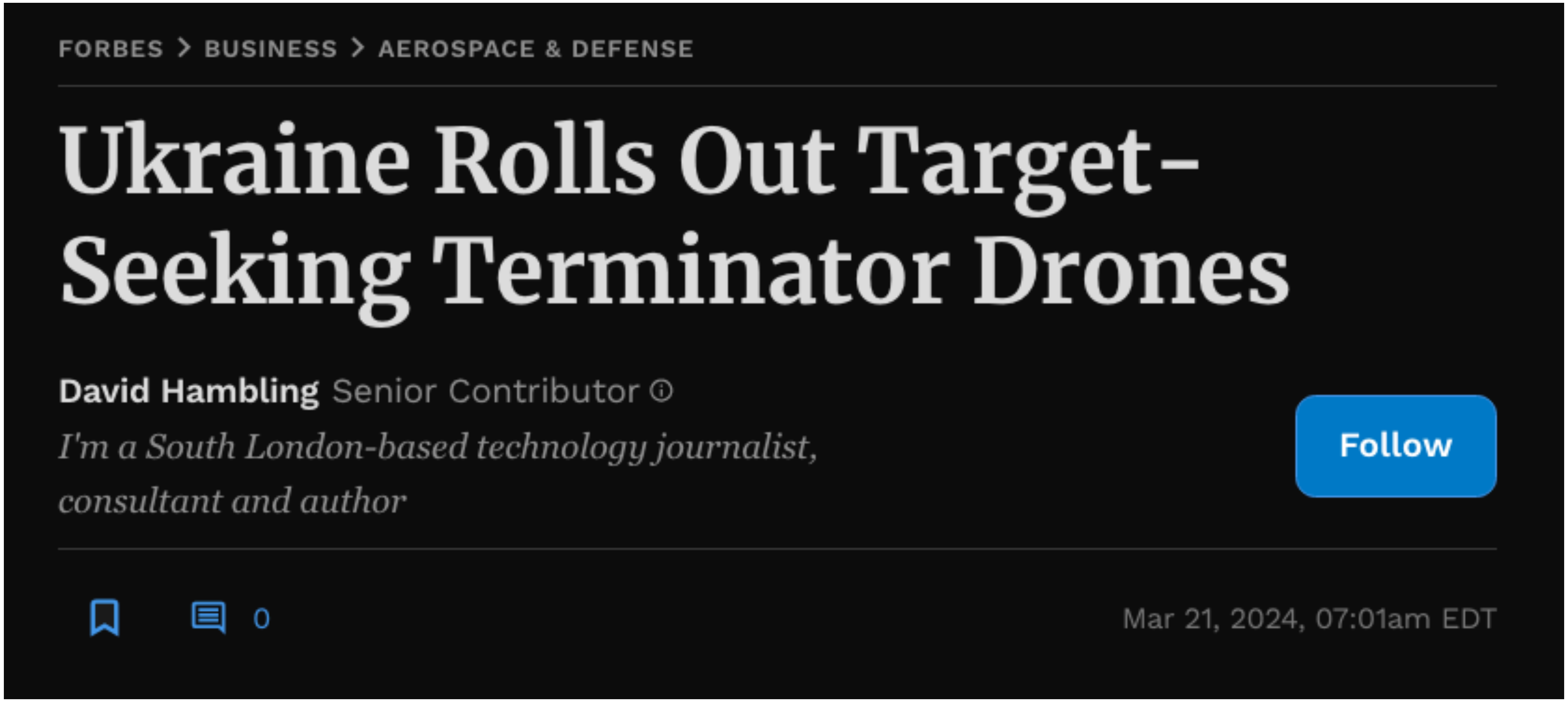
BY BILLY PERRIGO MARCH 11, 2024 9:00 AM EDT



Elon Musk predicts AI will be smarter than humans by next year

BY CHRIS MORRIS
April 9, 2024 at 7:39 AM PDT

FORTUNE

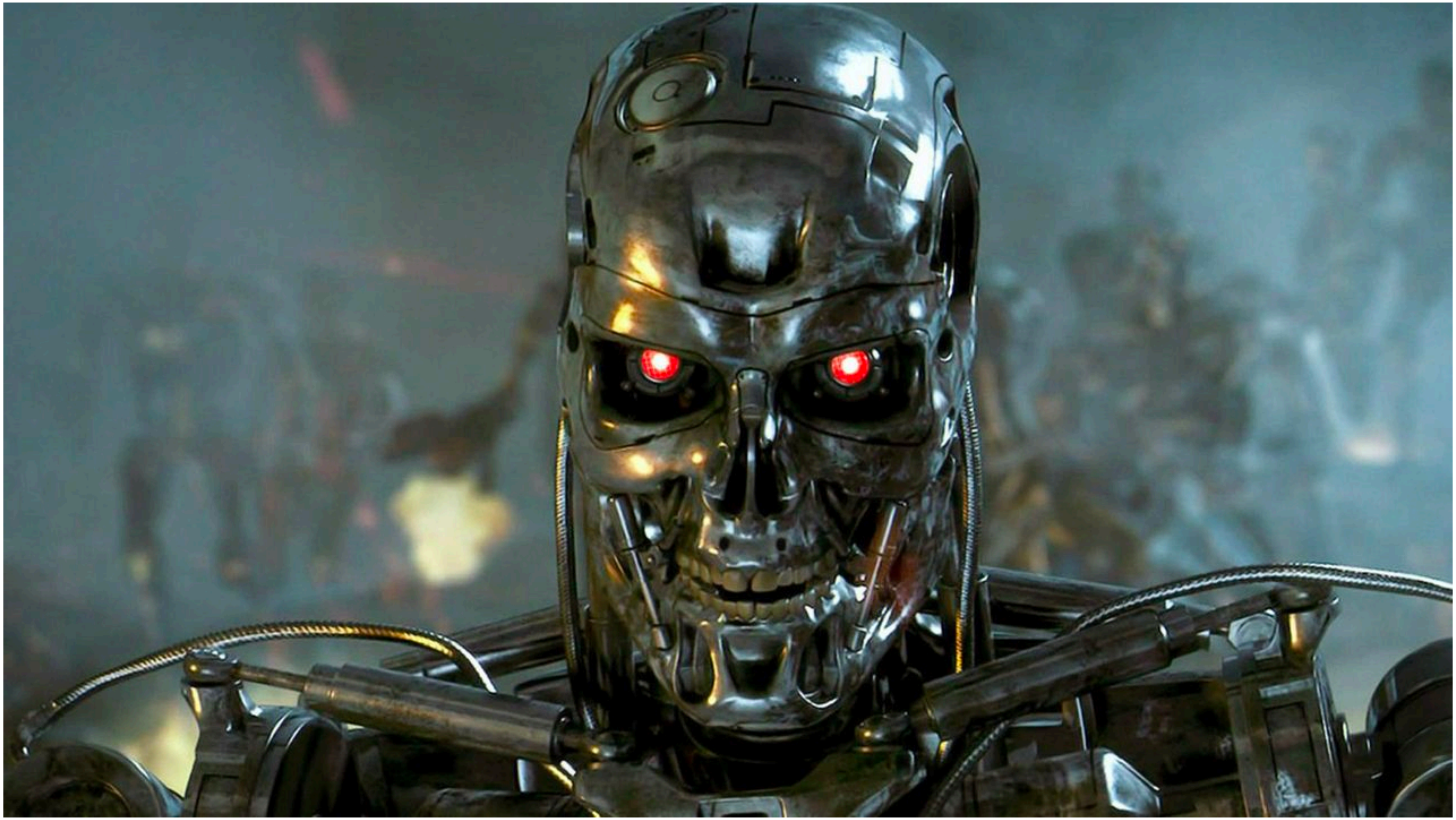


Ukraine Rolls Out Target-Seeking Terminator Drones

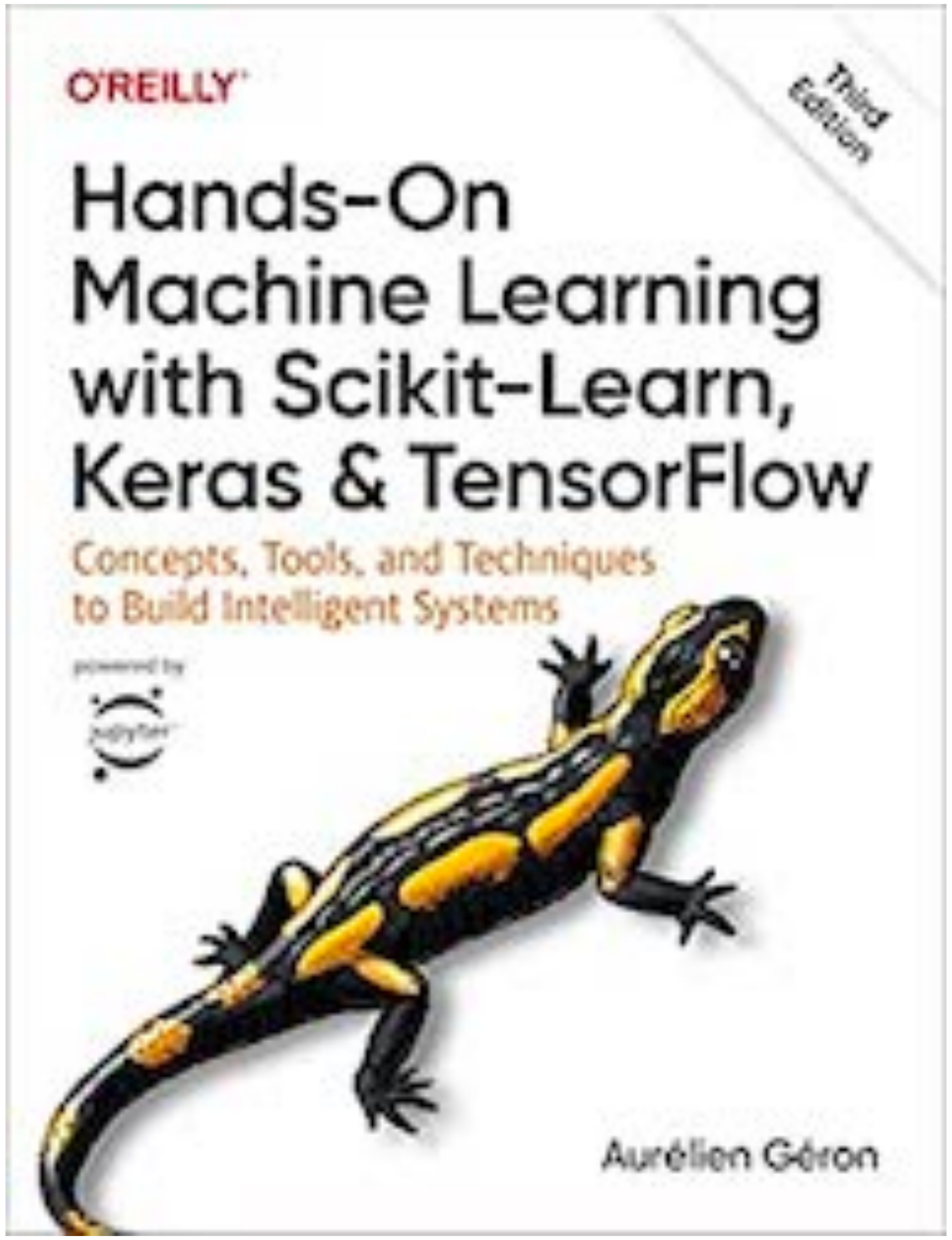
David Hambling Senior Contributor

I'm a South London-based technology journalist, consultant and author

Mar 21, 2024, 07:01am EDT



What Is Machine Learning?



O'REILLY

Third Edition

Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow

Concepts, Tools, and Techniques to Build Intelligent Systems

powered by jupyter

Aurélien Géron

What Is Machine Learning?

- The science (and art) of programming computers so they learn from data
- Example: **spam filter**
    - Learns from examples of spam emails, flagged by users, and regular emails (the *training set*)
    - The part of a ML system that learns and makes predictions is called a *model.*
    - Neural networks and random forests are examples of models

Why Use Machine Learning?

Traditional Programming

- To create a spam filter

  1. Examine spam examples and find words that appear often, like "4U", "credit card", "free", "amazing", or other paterns in sender's name or email body

  2. Write an algorithm to detect each pattern, flag email as spam if a number of these patterns are detected

  3. Test the program and repeat steps 1 and 2 until it's good enough to launch

Traditional Programming

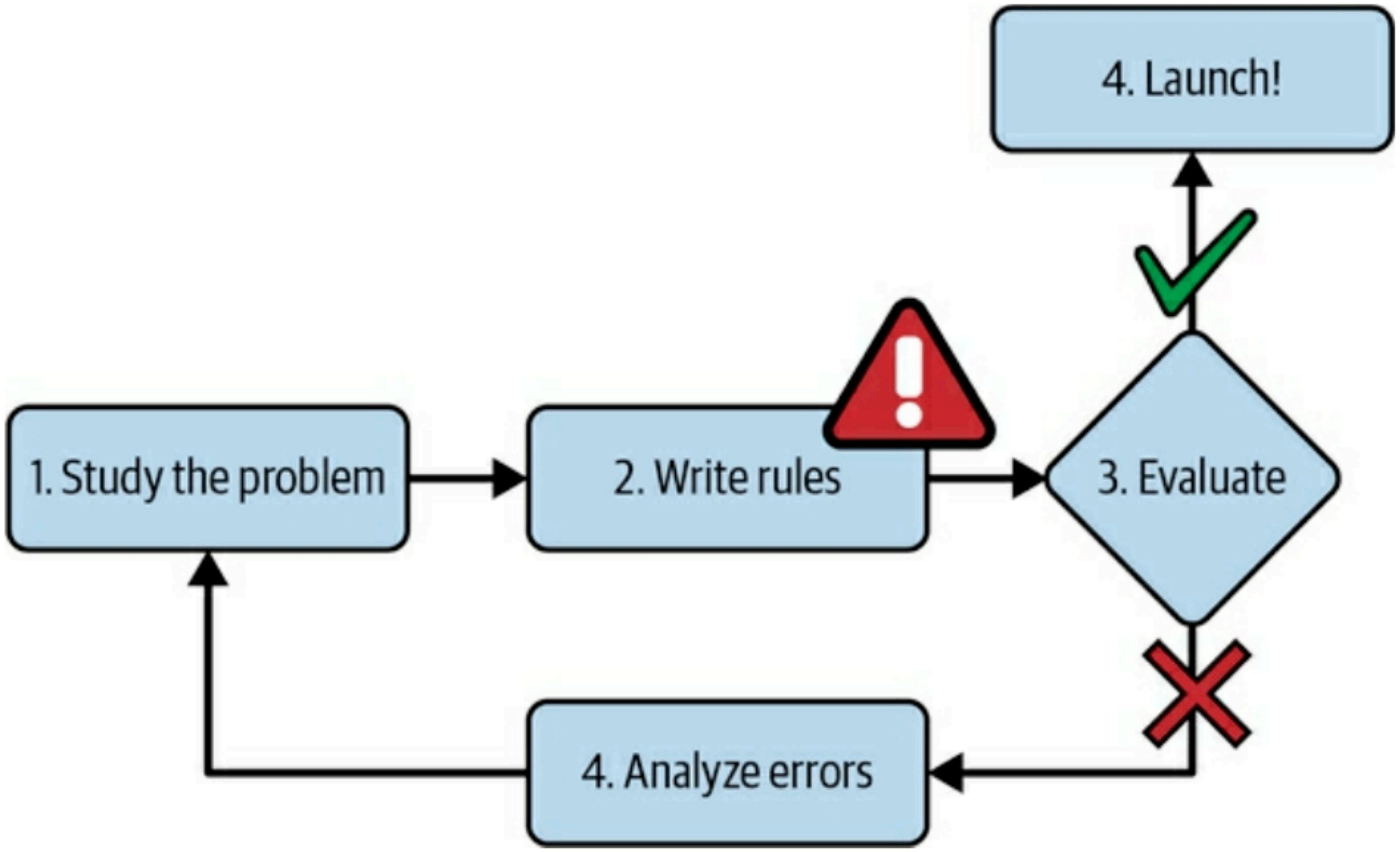


Figure 1-1. The traditional approach

- Program uses a long list of rules
- Difficult to maintain

Machine Learning

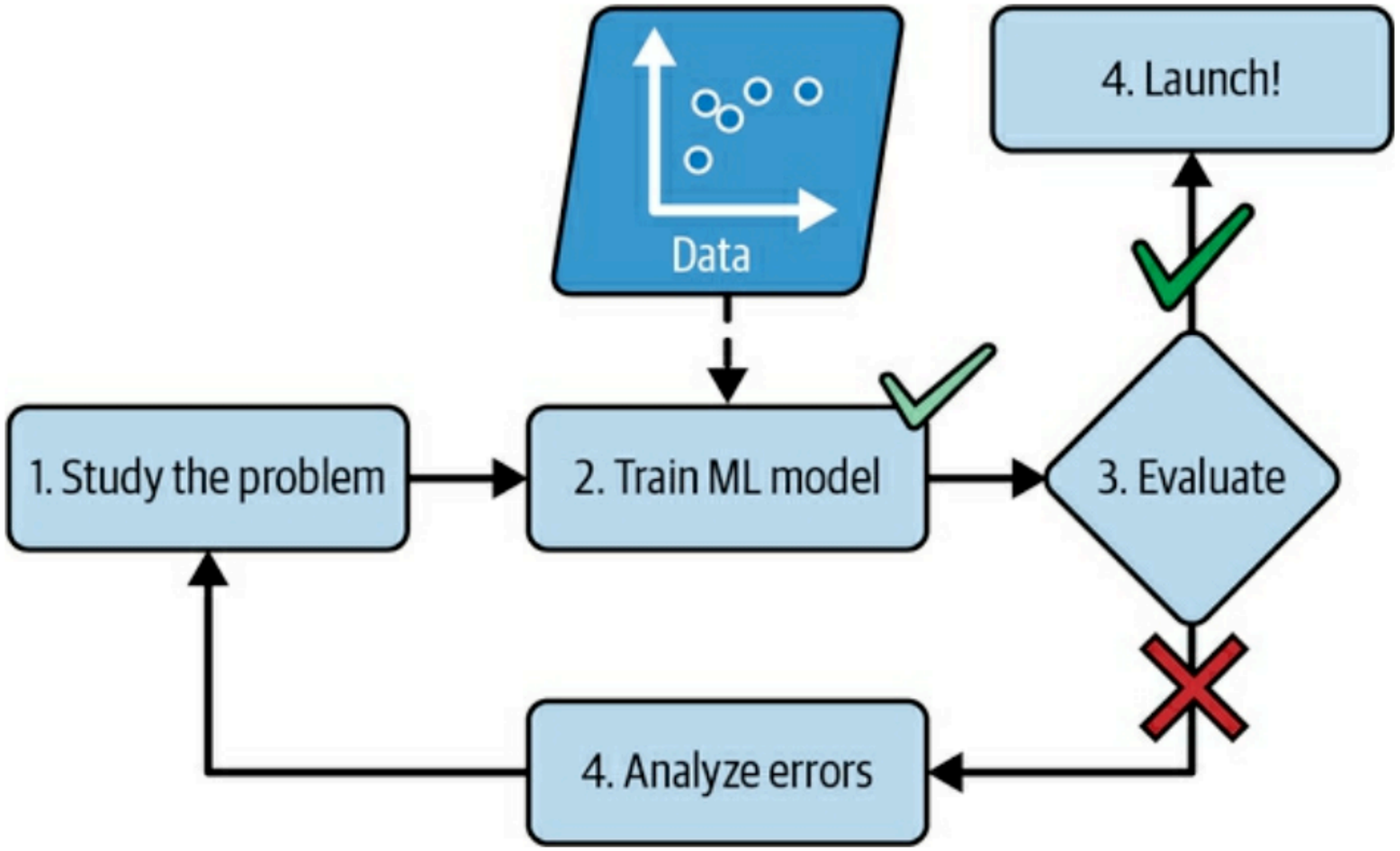


Figure 1-2. The machine learning approach

- Learns words and phrases that can predict spam
- Program is shorter, easier to maintain, and more accurate

Machine Learning

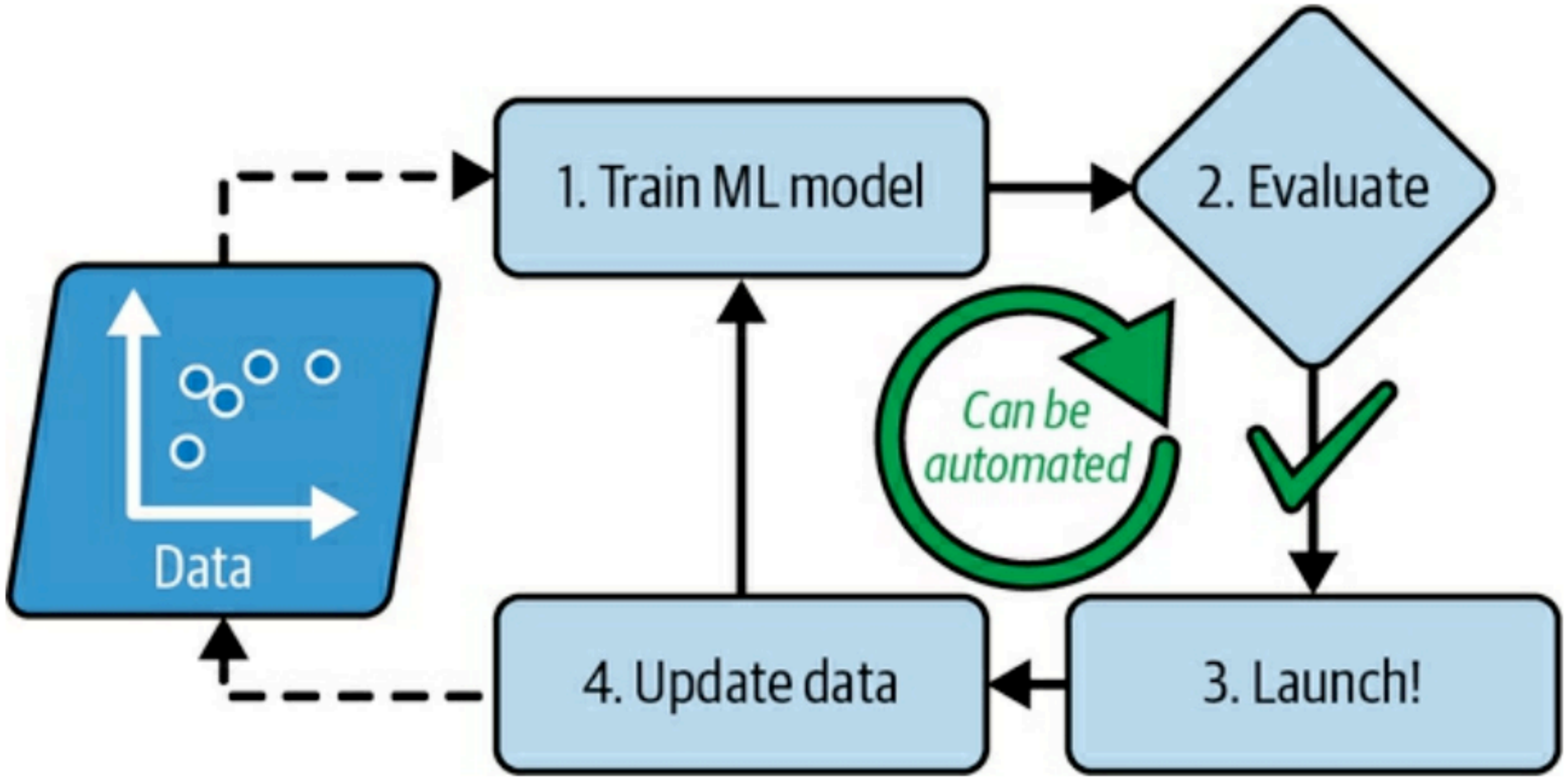


Figure 1-3. Automatically adapting to change

- Automatically adapts to changing tactics of spammers

Other Problems

- Some problems are too complex for traditional approaches
  - Or have no known algorithm
- **Speech recognition**
  - Trained on many example recordings

Examples of Applications

Examples of Applications

- Analyzing images of products on a production line to automatically classify them
  - Image classification typically uses **Convolutional Neural Networks (CNNs)** or **Transformers**
- Detecting tumors in brain scans
- Automatically classifying news articles
  - This is **Natural Language Processing (NLP)**, and more specifically text classification, which can be tackled using **Recurrent Neural Networks (RNNs)** and **CNNs**, but **Transformers** work even better

Examples of Applications

- Creating a chatbot or a personal assistant
- Forecasting your company’s revenue next year, based on many performance metrics
- Making your app react to voice commands
- Detecting credit card fraud
- Segmenting clients based on their purchases so that you can design a different marketing strategy for each segment
- Representing a complex, high-dimensional dataset in a clear and insightful diagram
- Recommending a product that a client may be interested in, based on past purchases
- Building an intelligent bot for a game

Types of Machine Learning Systems

Supervised Learning

- Training data has labels
  - Indicating desired solution

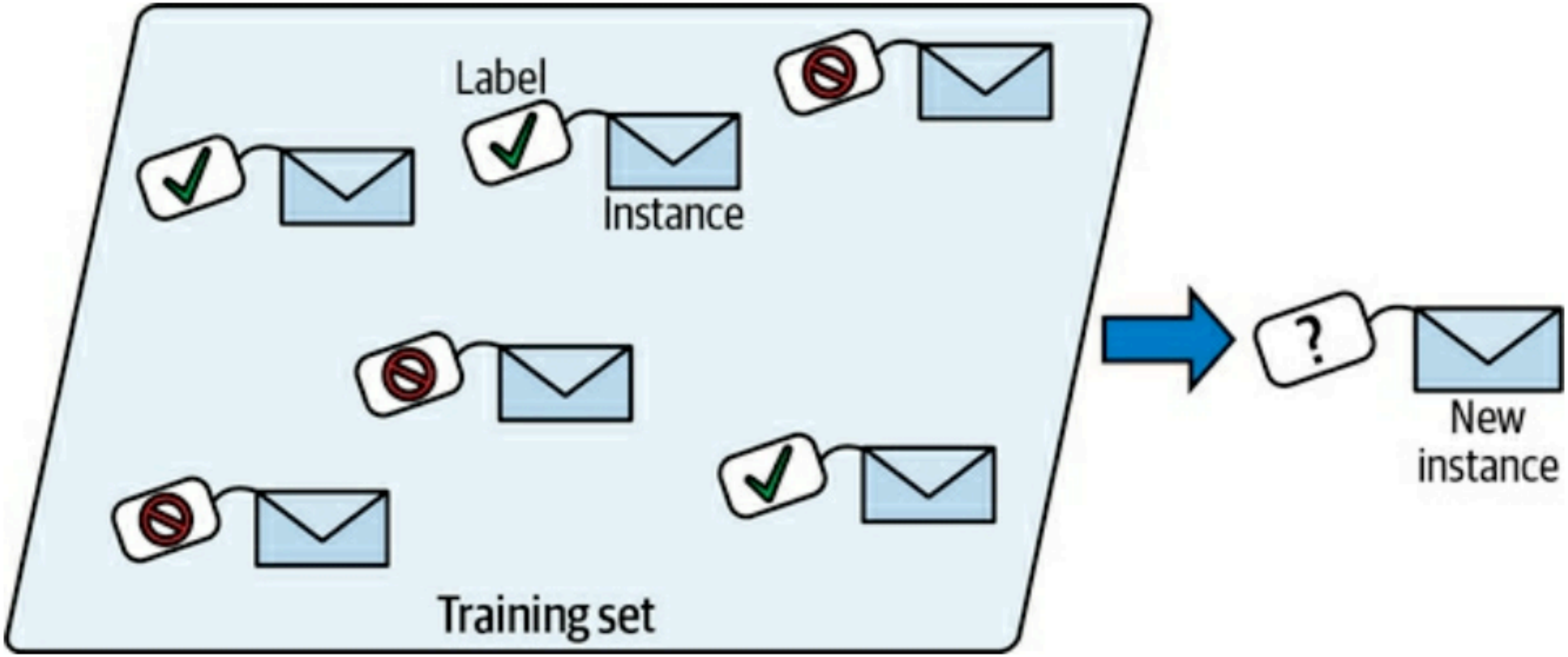


Figure 1-5. A labeled training set for spam classification (an example of supervised learning)

Supervised Learning Tasks

- **Classification**
  - Sort input samples into categories, like a spam filter
- **Regression**
  - Predict a target numerical value, like the price of a car, given a set of features, like mileage, age, brand, etc.

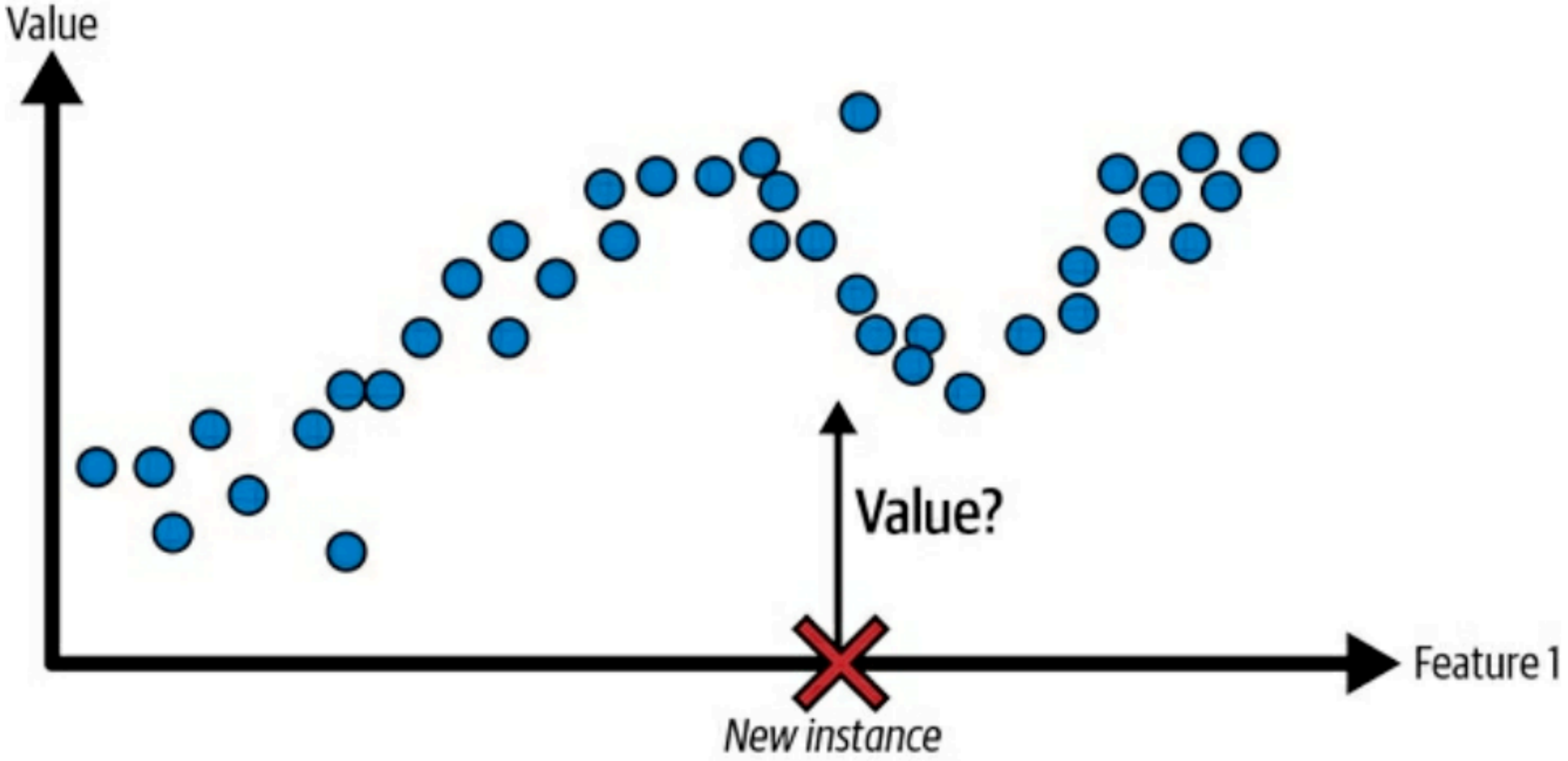


Figure 1-6. A regression problem: predict a value, given an input feature (there are usually multiple input features, and sometimes multiple output values)

Unsupervised Learning

- Training data is unlabeled

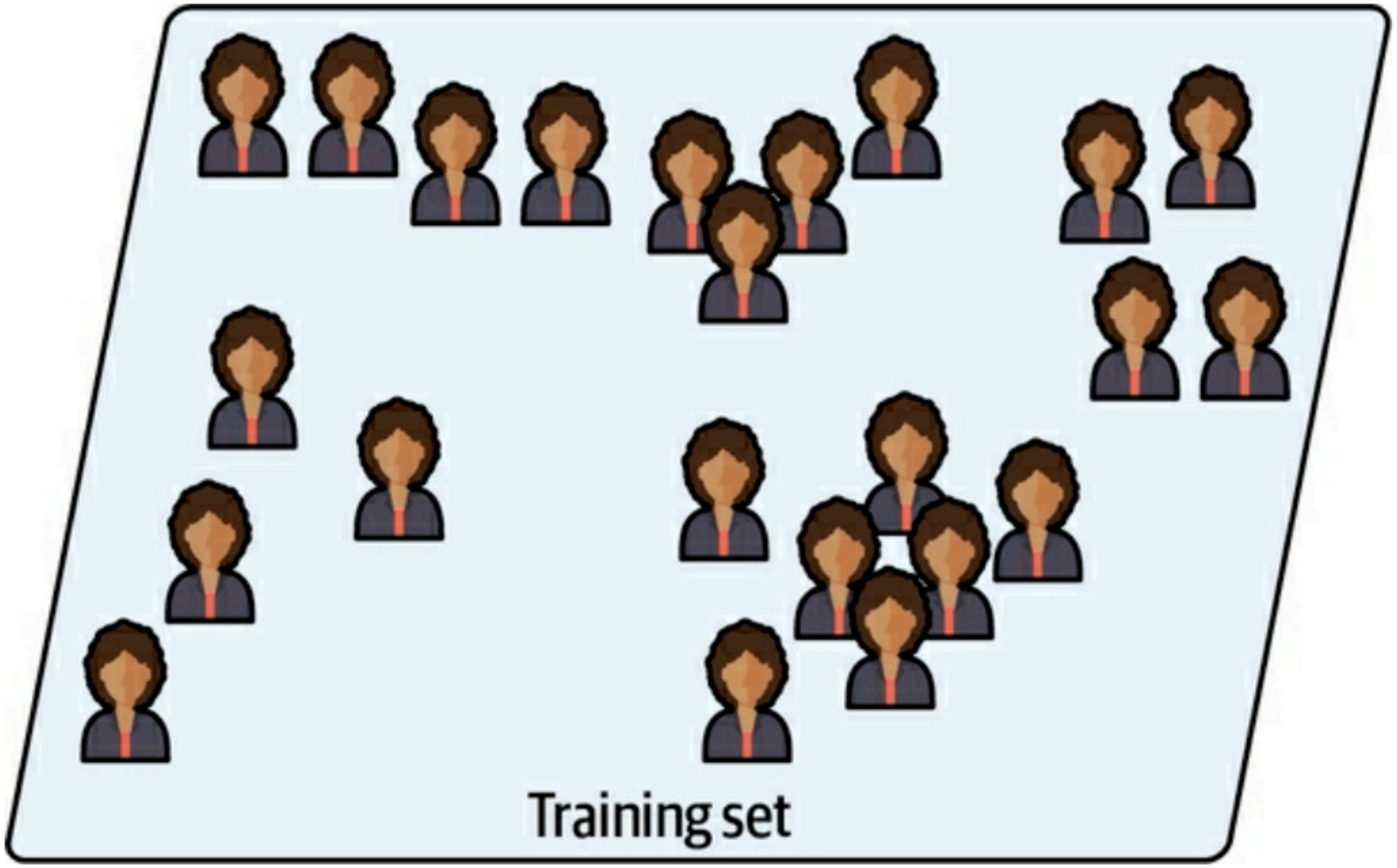


Figure 1-7. An unlabeled training set for unsupervised learning

Unsupervised Learning

- **Clustering** algorithm
  - Sorts data into groups

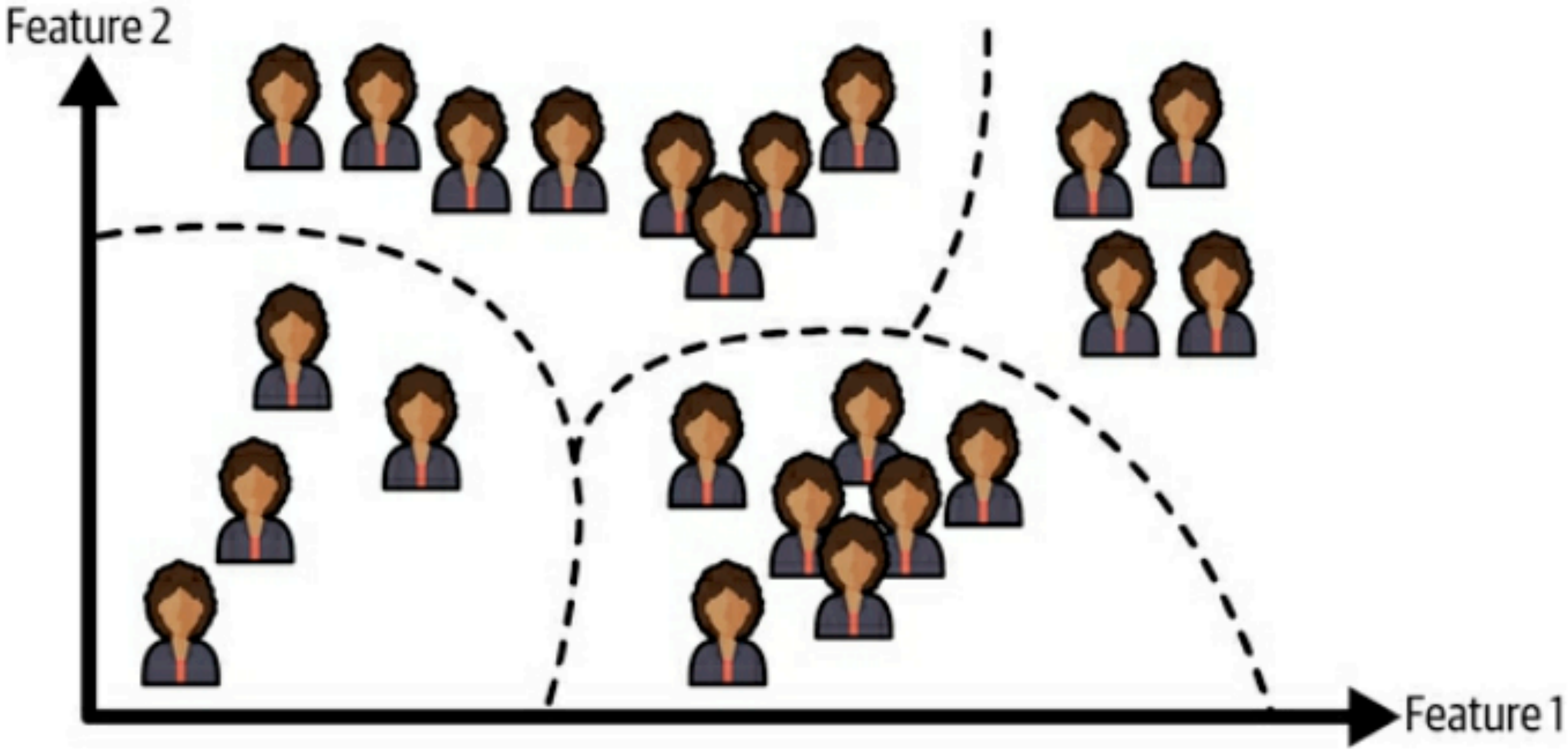


Figure 1-8. Clustering

Unsupervised Learning

- **Visualization**

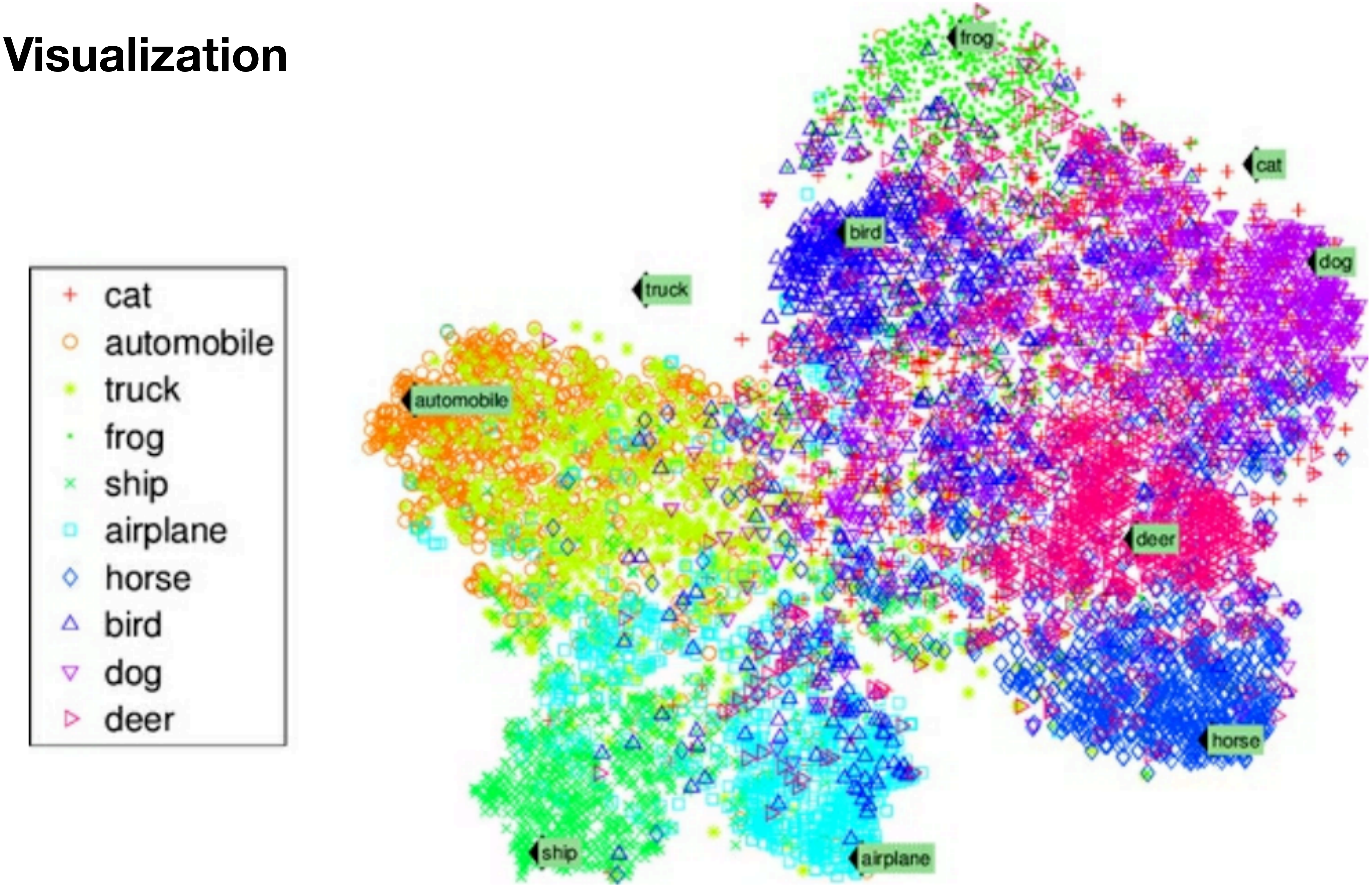


Figure 1-9. Example of a t-SNE visualization highlighting semantic clusters[2]

Dimensionality Reduction

- A way to simplify data
- without losing too much information
- Example: merge correlated features into one
  - For a car, combine *milage* and *age* into *wear-and-tear*
  - This is called *feature extraction*

Unsupervised Learning

- **Anomaly detection**
  - Find unusual credit card transactions
  - Find manufacturing defects

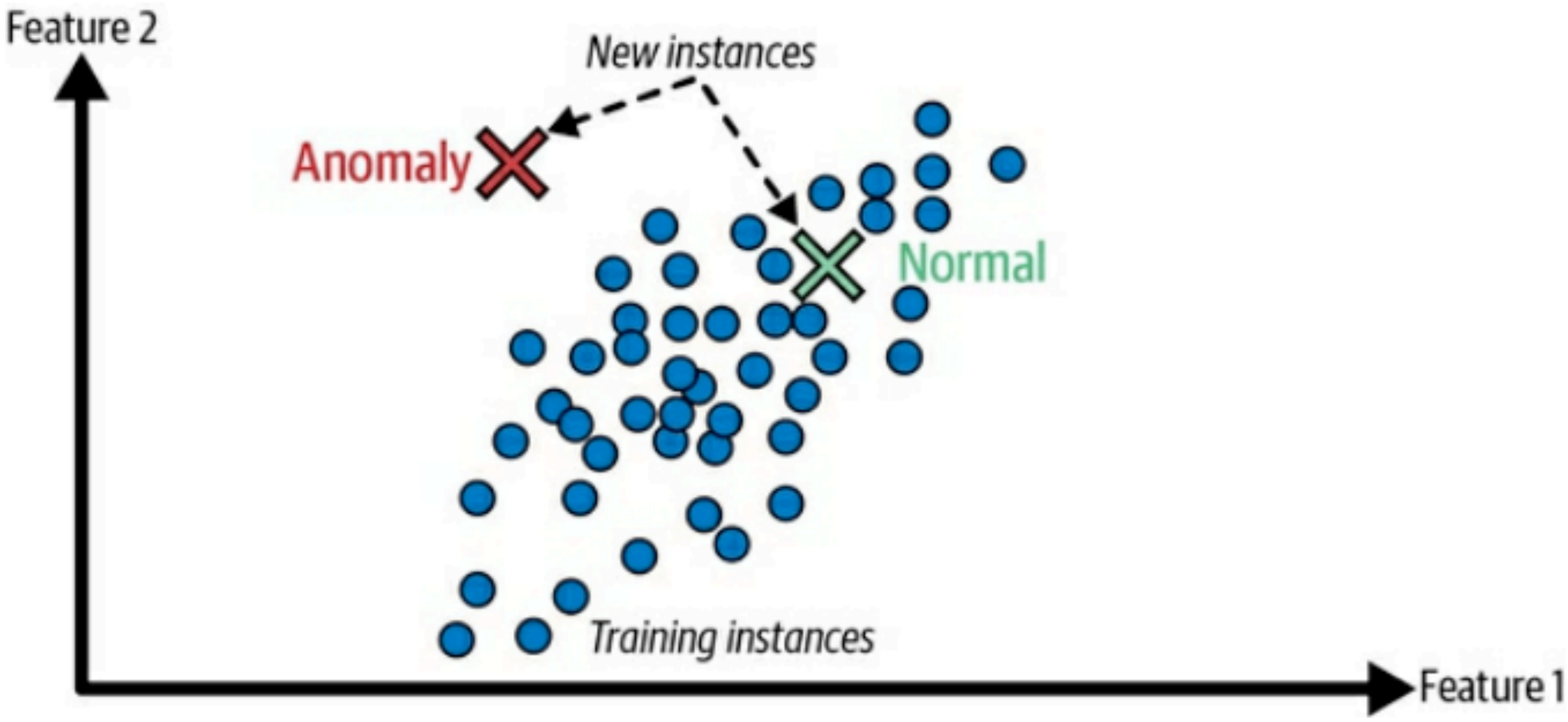


Figure 1-10. Anomaly detection

Unsupervised Learning

- **Association rule learning**
  - Discover interesting relations between attributes
  - Find items customers purchase together

Semi-Supervised Learning

- First unsupervised model groups similar images together
- Then it asks the user to label a group at a time

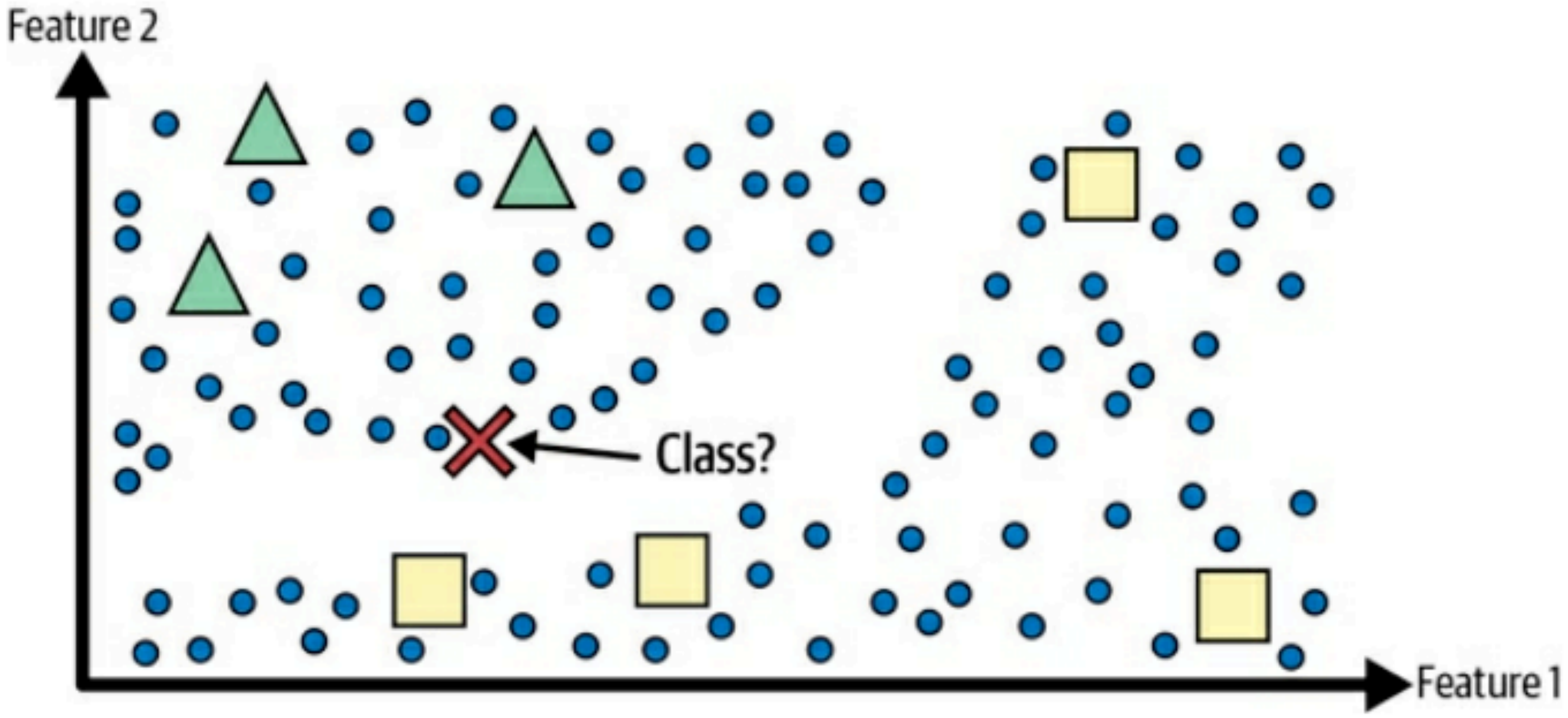


Figure 1-11. Semi-supervised learning with two classes (triangles and squares): the unlabeled examples (circles) help classify a new instance (the cross) into the triangle class rather than the square class, even though it is closer to the labeled squares

Self-Supervised Learning

- Generates a labeled dataset from an unlabeled one
  - Example: mask part of an image, train a model to recover the original image

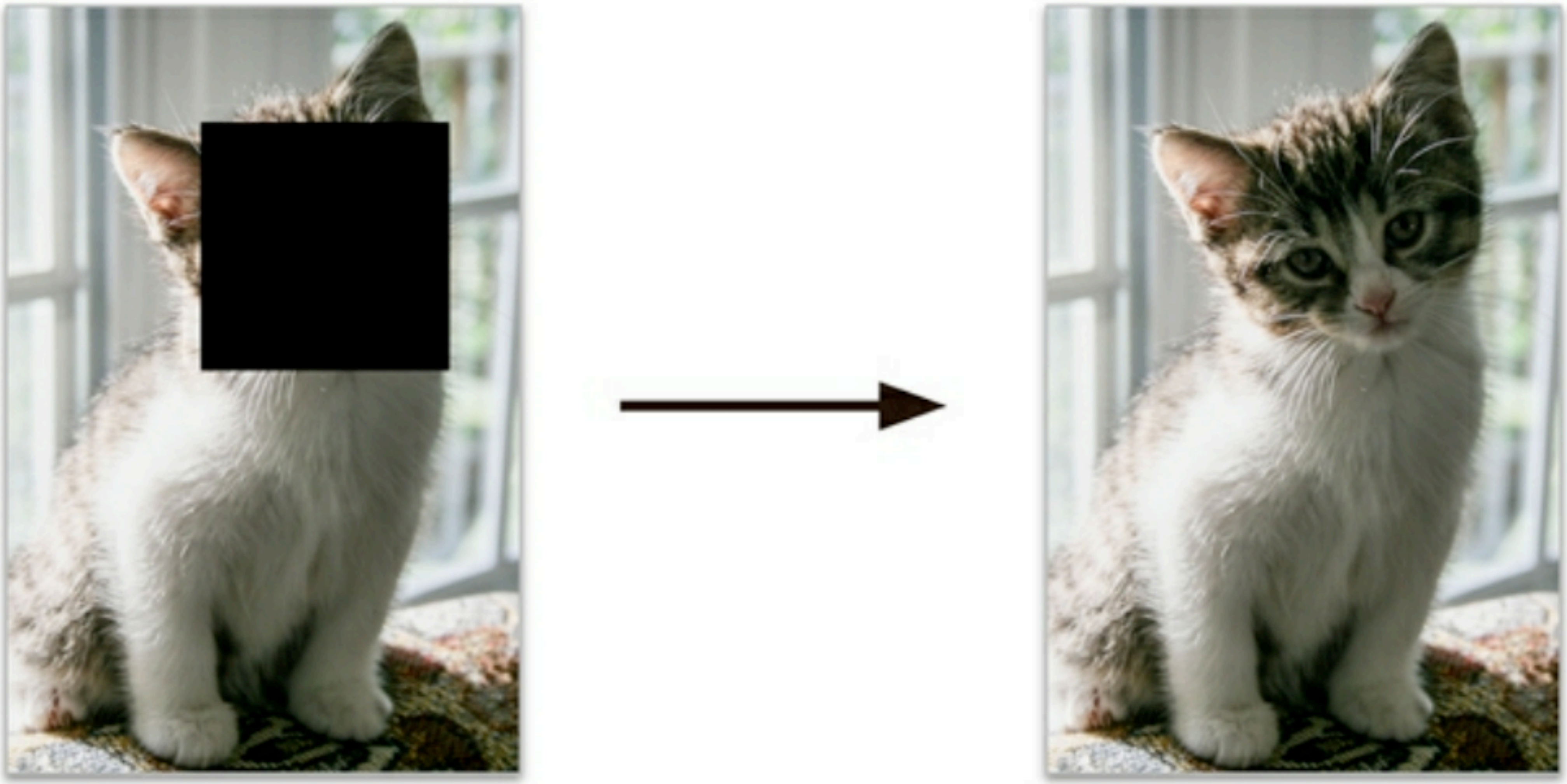


Figure 1-12. Self-supervised learning example: input (left) and target (right)

Reinforcement Learning

- The learning system, called an **agent**
  - Observes the environment
  - Selects and performs actions
  - Gets **rewards** or **penalties**
- Like a robot learning to walk

Reinforcement Learning

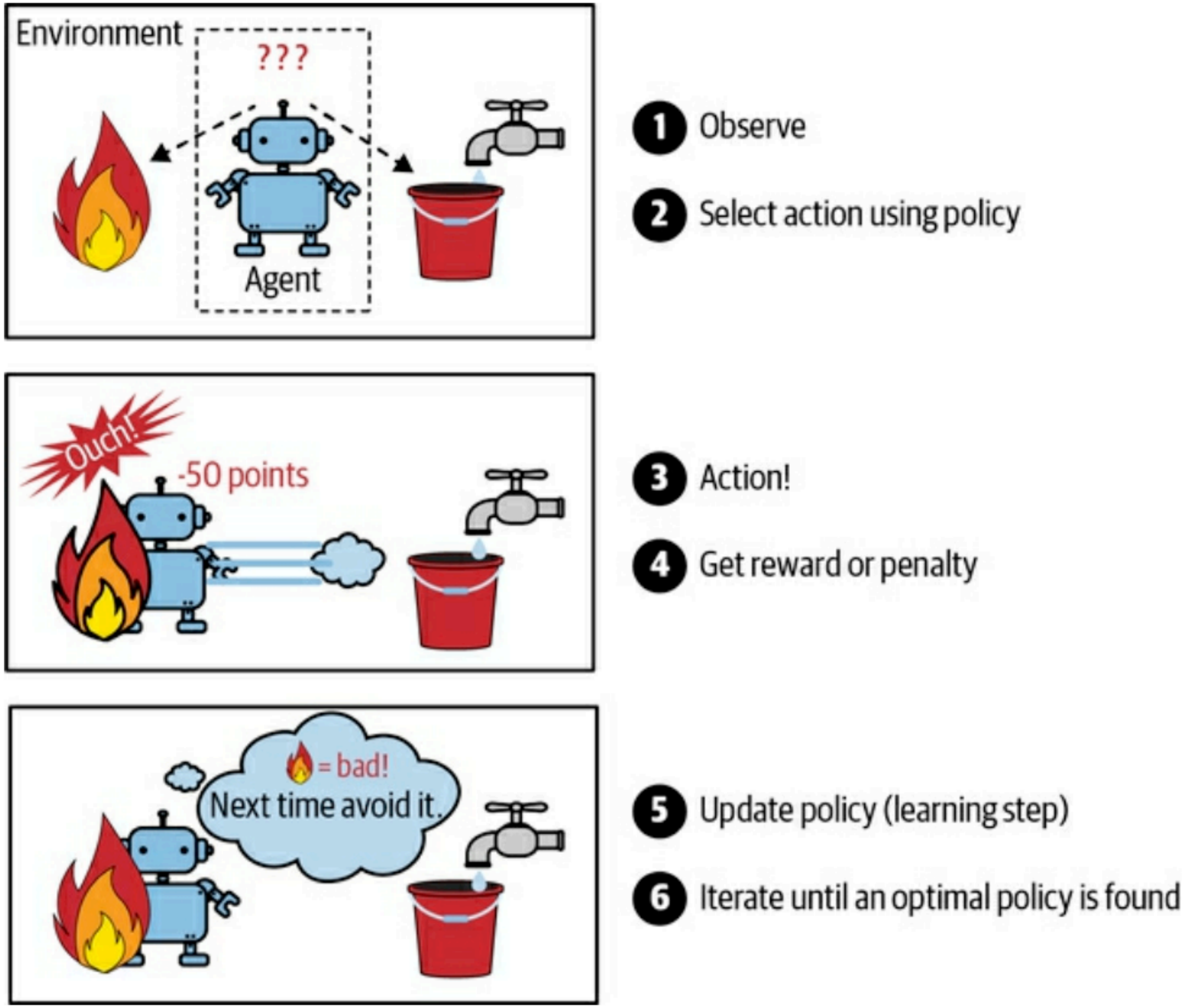

Environment
???
Agent
Ouch!
-50 points
= bad!
Next time avoid it.
1 Observe
2 Select action using policy
3 Action!
4 Get reward or penalty
5 Update policy (learning step)
6 Iterate until an optimal policy is found

Figure 1-13. Reinforcement learning

Securing AI Systems

April 12, 2024

NIST AI 100-1

Artificial Intelligence Risk Management Framework (AI RMF 1.0)

NIST NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
U.S. DEPARTMENT OF COMMERCE

Harm to People

- Individual: Harm to a person's civil liberties, rights, physical or psychological safety, or economic opportunity.
- Group/Community: Harm to a group such as discrimination against a population sub-group.
- Societal: Harm to democratic participation or educational access.

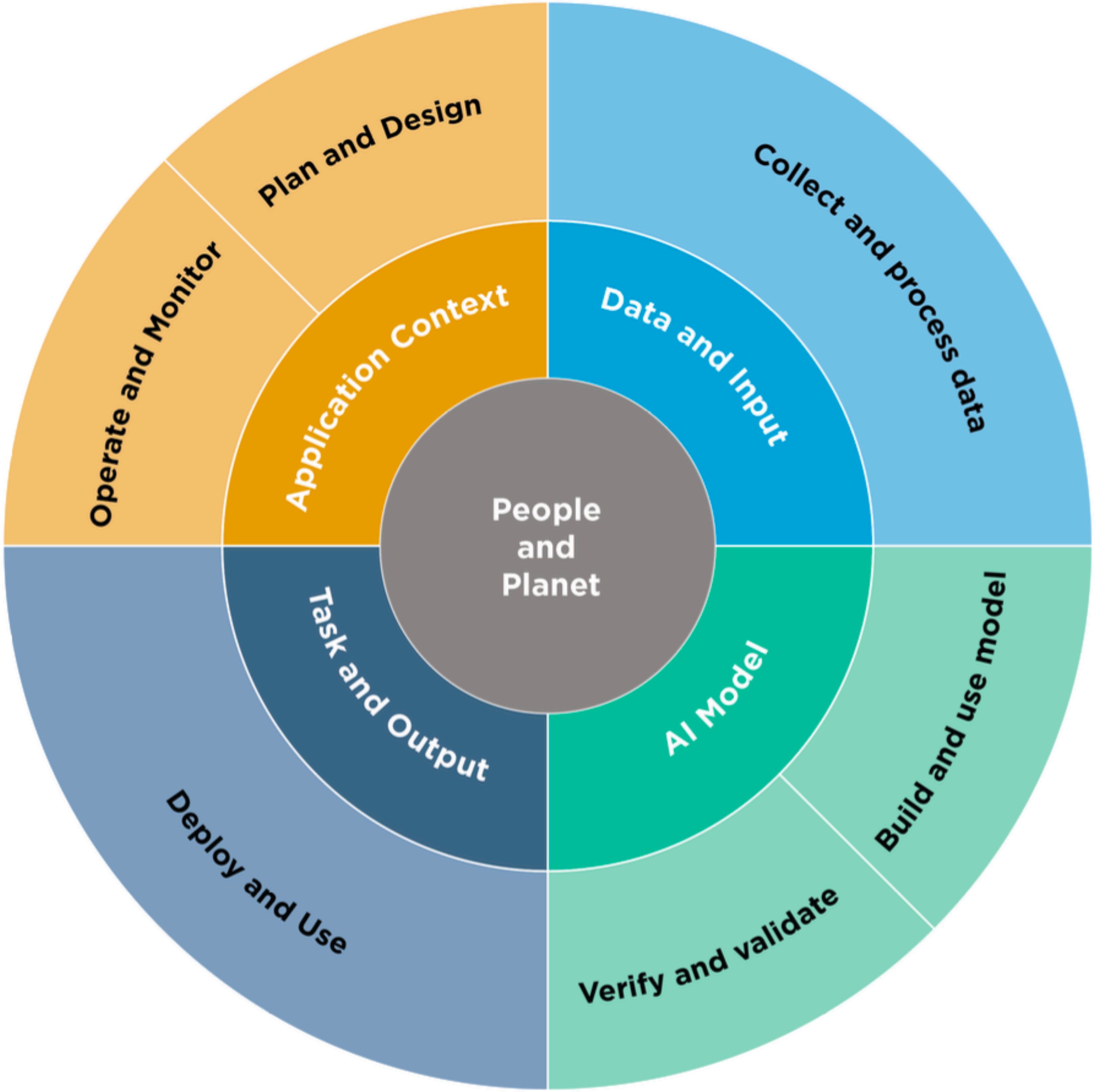
Harm to an Organization

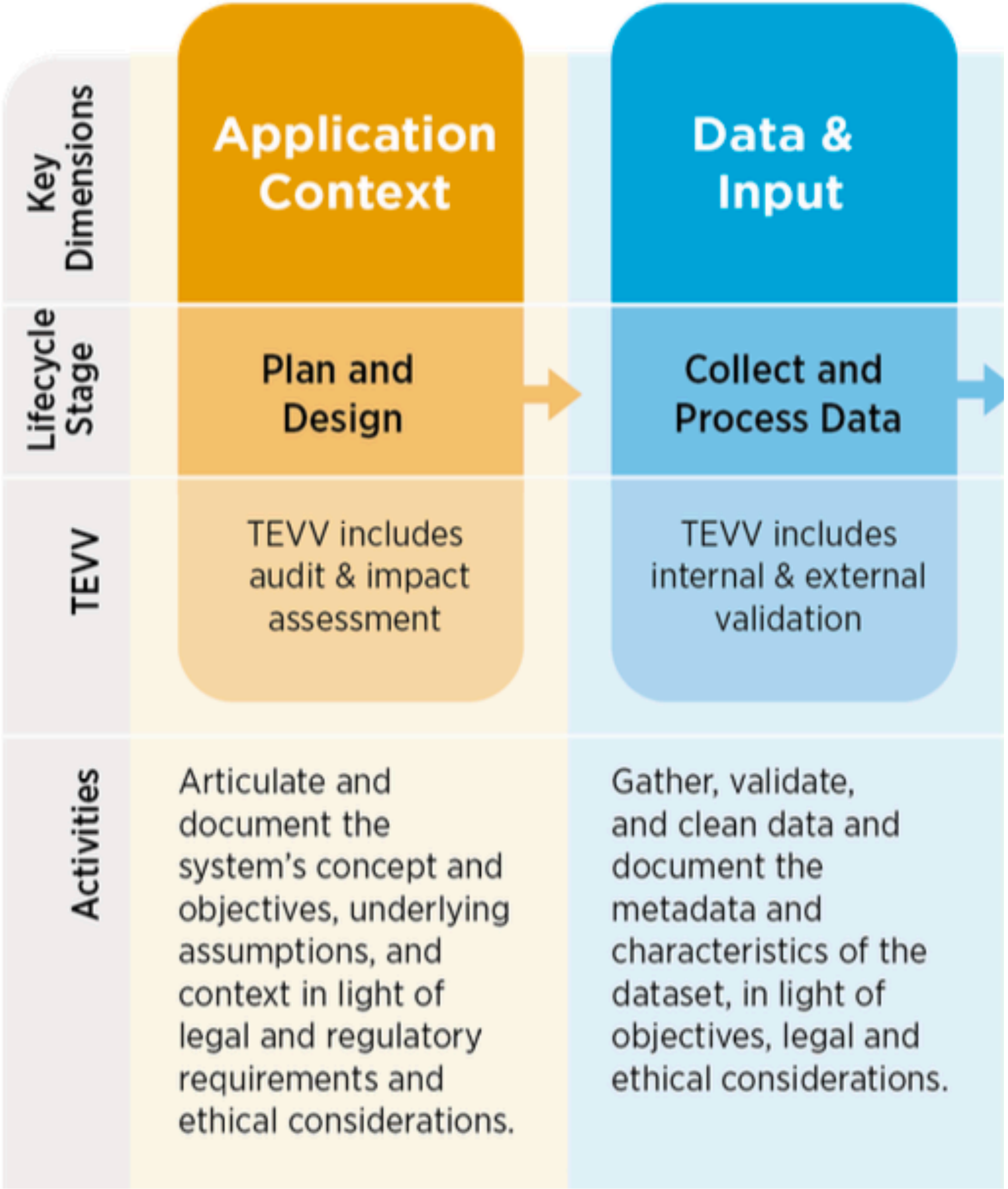
- Harm to an organization's business operations.
- Harm to an organization from security breaches or monetary loss.
- Harm to an organization's reputation.

Harm to an Ecosystem

- Harm to interconnected and interdependent elements and resources.
- Harm to the global financial system, supply chain, or interrelated systems.
- Harm to natural resources, the environment, and planet.

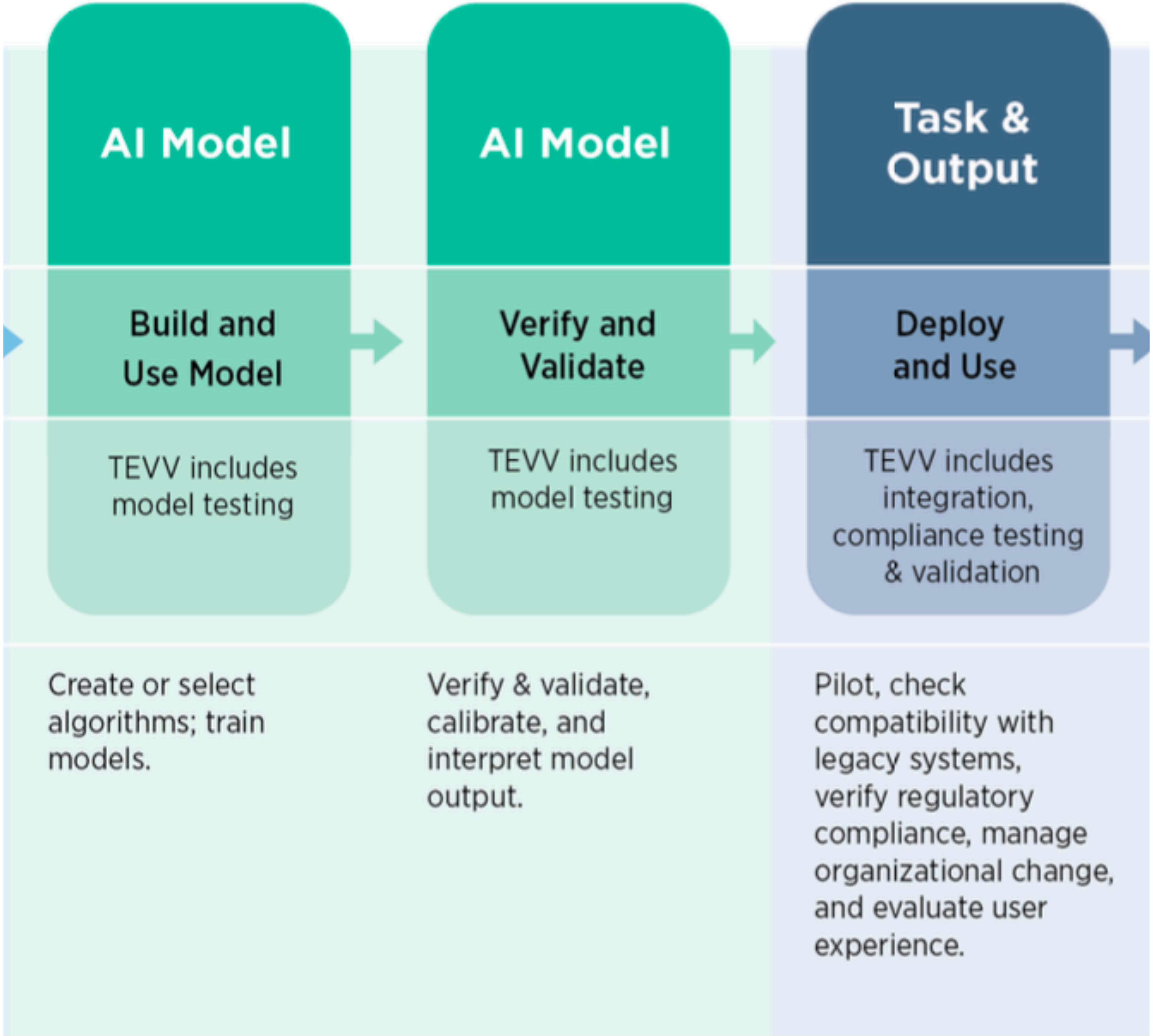
Lifecycle and Key Dimensions





Key Dimensions	Application Context	Data & Input
Lifecycle Stage	Plan and Design →	Collect and Process Data →
TEVV	TEVV includes audit & impact assessment	TEVV includes internal & external validation
Activities	Articulate and document the system's concept and objectives, underlying assumptions, and context in light of legal and regulatory requirements and ethical considerations.	Gather, validate, and clean data and document the metadata and characteristics of the dataset, in light of objectives, legal and ethical considerations.

TEVV: Test, Evaluation, Verification, and Validation



AI Model

Build and Use Model

TEVV includes model testing

Create or select algorithms; train models.

AI Model

Verify and Validate

TEVV includes model testing

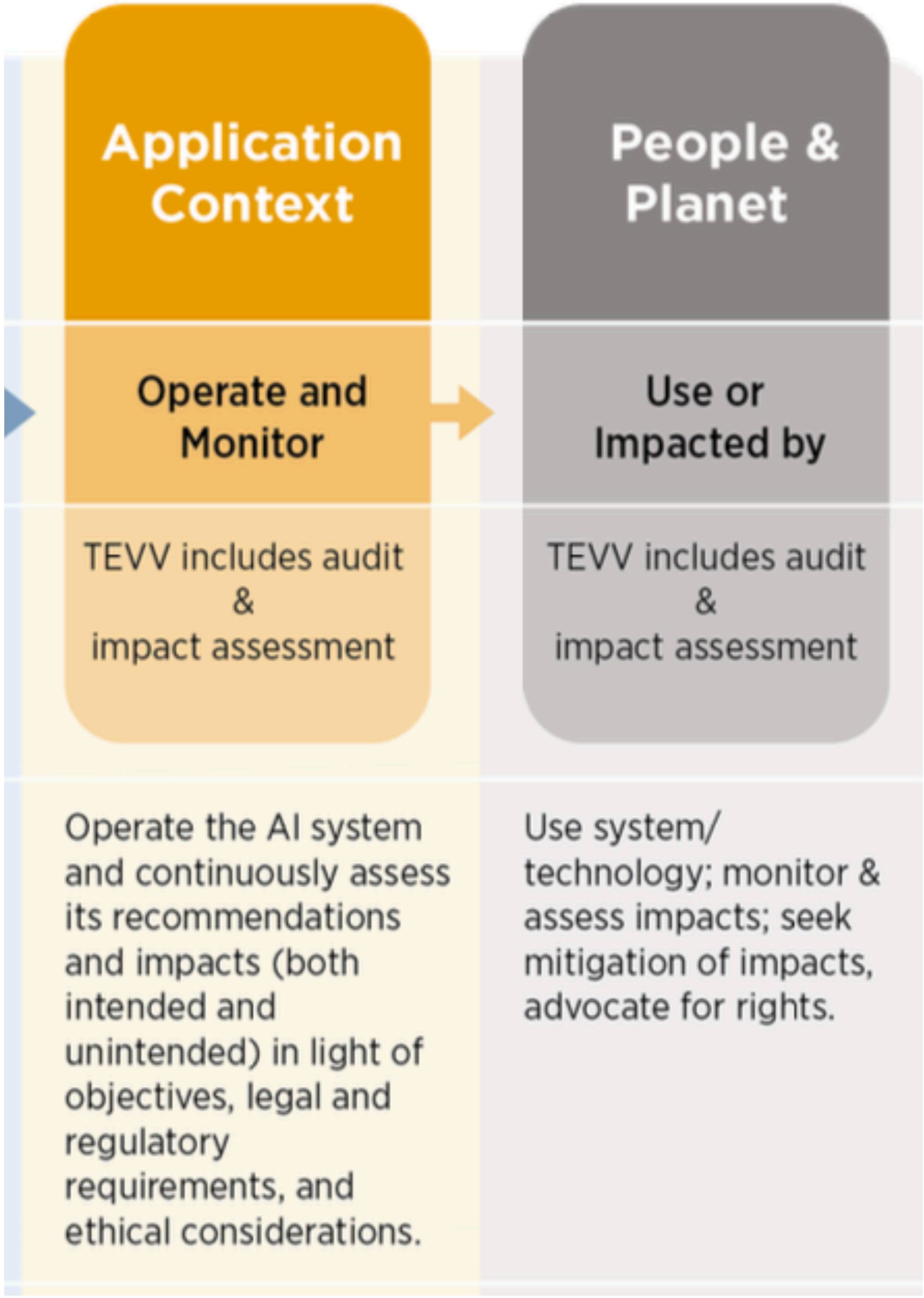
Verify & validate, calibrate, and interpret model output.

Task & Output

Deploy and Use

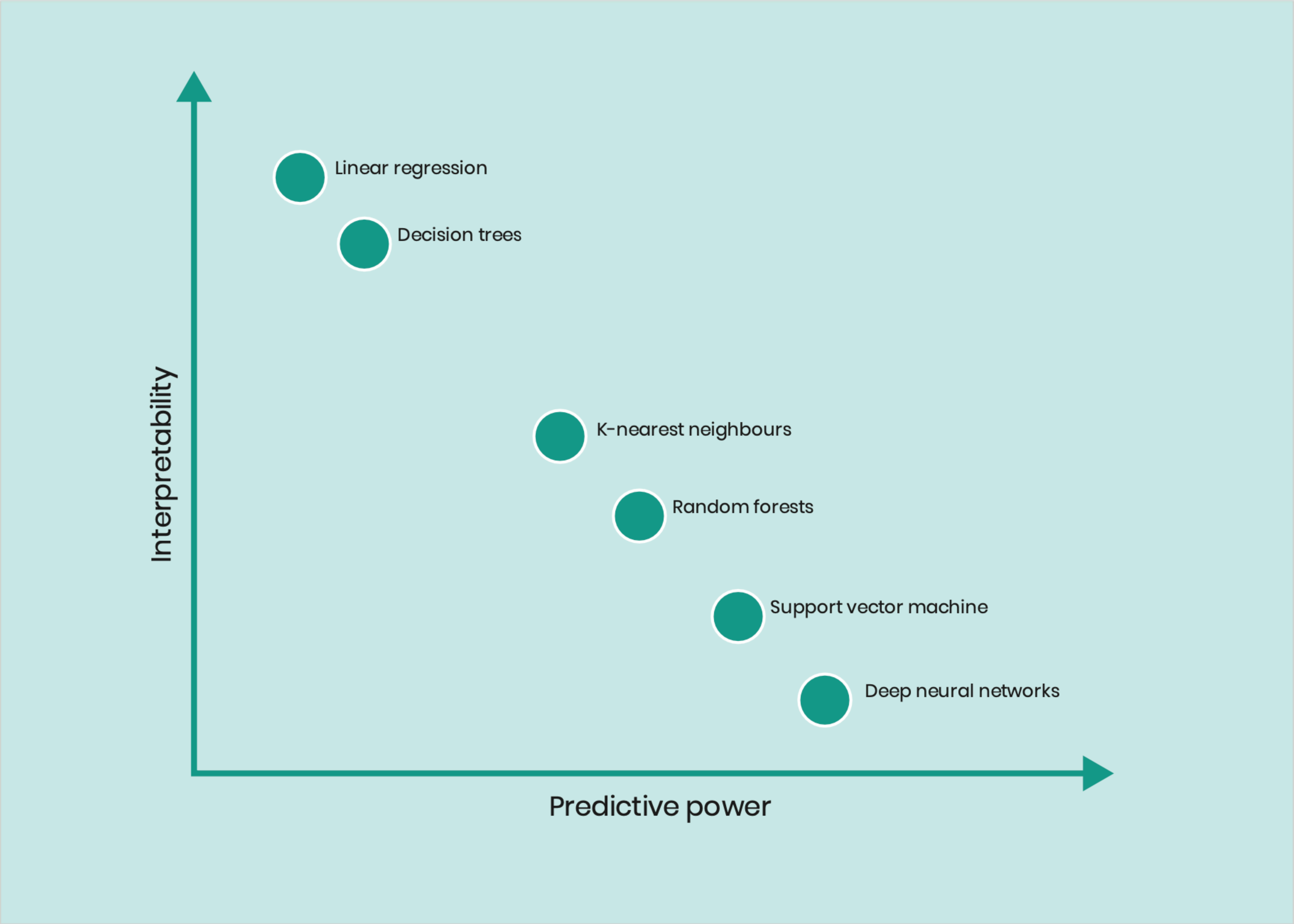
TEVV includes integration, compliance testing & validation

Pilot, check compatibility with legacy systems, verify regulatory compliance, manage organizational change, and evaluate user experience.

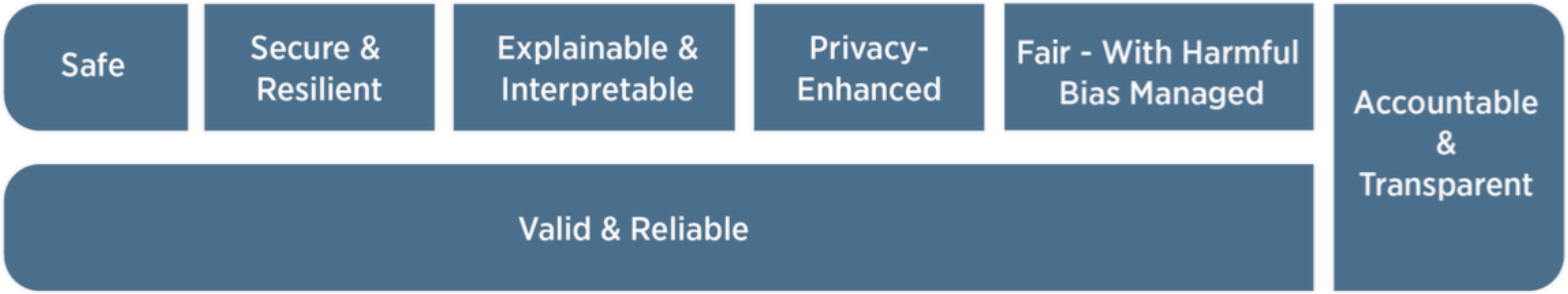


Application Context	People & Planet
Operate and Monitor	Use or Impacted by
TEVV includes audit & impact assessment	TEVV includes audit & impact assessment
Operate the AI system and continuously assess its recommendations and impacts (both intended and unintended) in light of objectives, legal and regulatory requirements, and ethical considerations.	Use system/ technology; monitor & assess impacts; seek mitigation of impacts, advocate for rights.

Inscrutability



Characteristics of Trustworthy AI



Safe

Secure & Resilient

Explainable & Interpretable

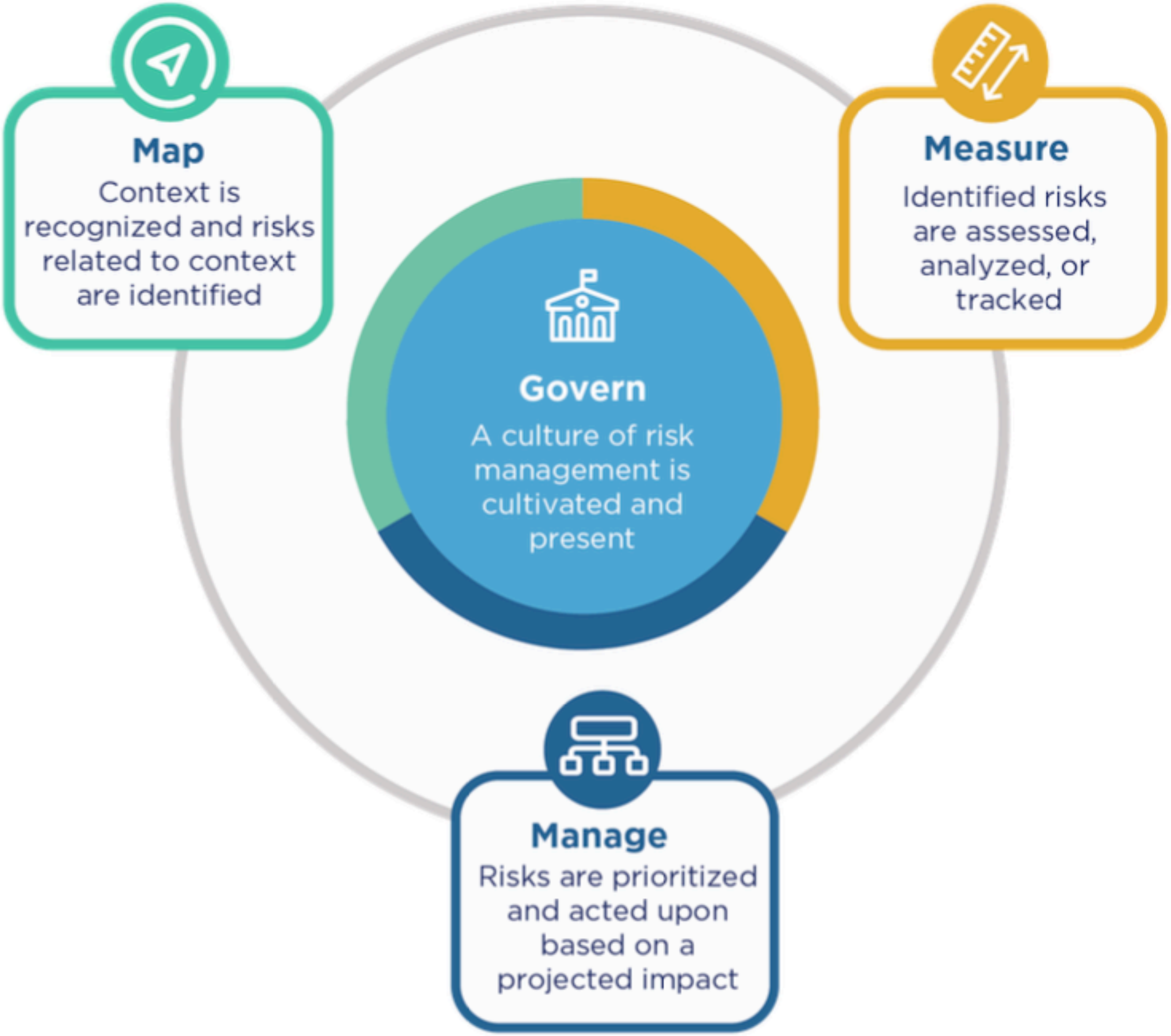
Privacy-Enhanced

Fair - With Harmful Bias Managed

Accountable & Transparent

Valid & Reliable

AI RMF Core



Map
Context is recognized and risks related to context are identified

Measure
Identified risks are assessed, analyzed, or tracked

Govern
A culture of risk management is cultivated and present

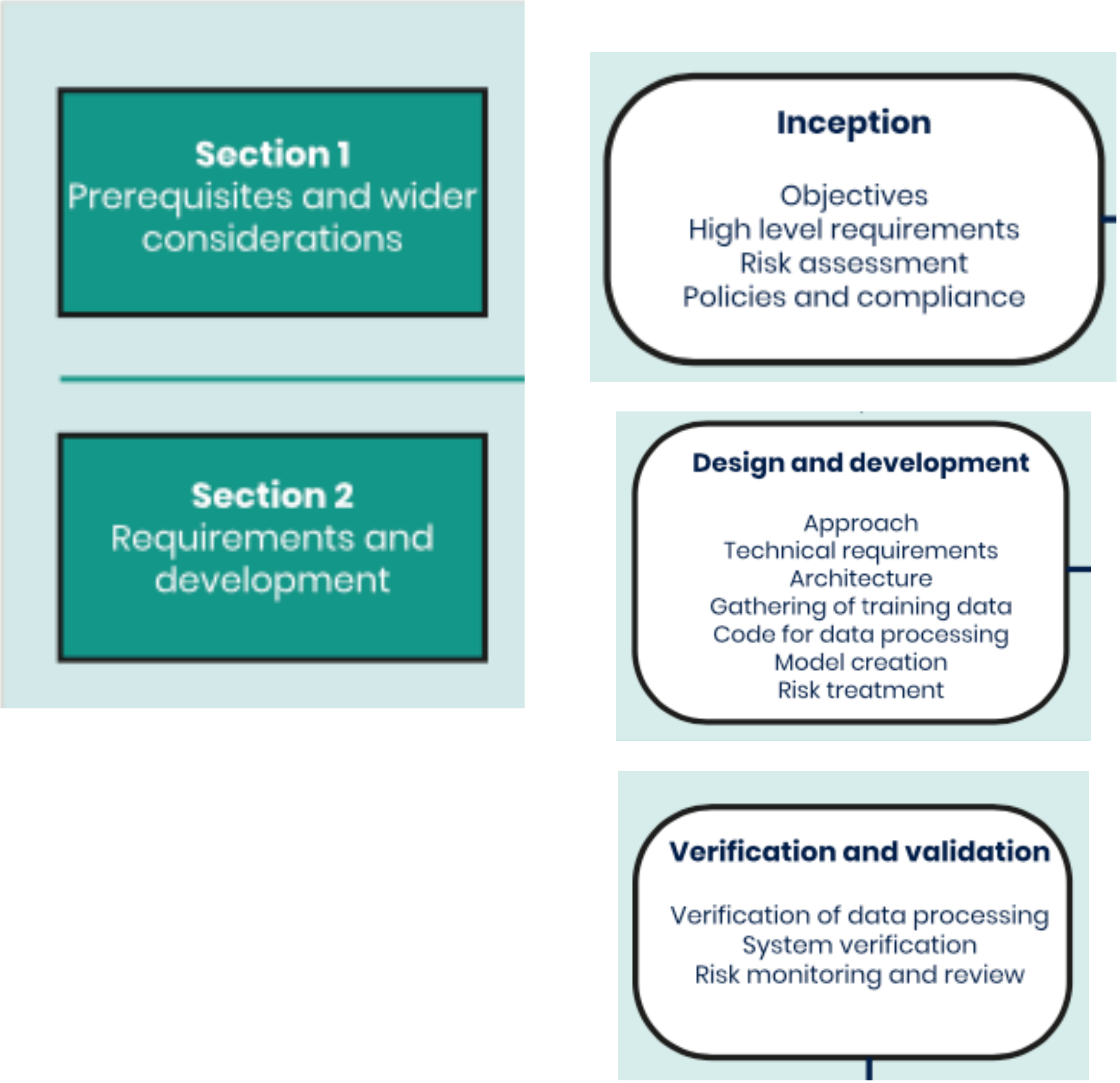
Manage
Risks are prioritized and acted upon based on a projected impact

Principles for the security of machine learning

Version 1

Published August 2022

National Cyber Security Centre
a part of GCHQ



Section 1
Prerequisites and wider considerations

Section 2
Requirements and development

Inception

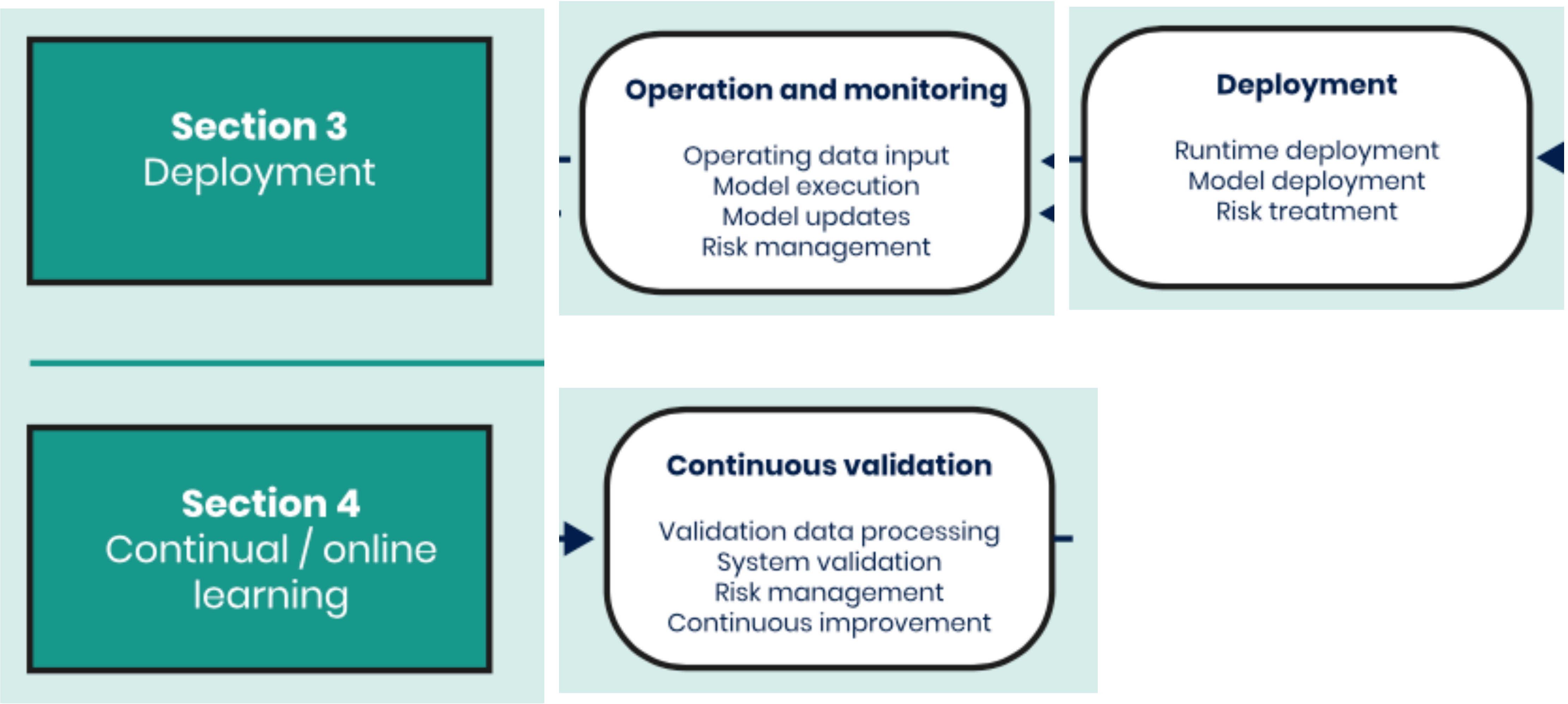
Objectives
High level requirements
Risk assessment
Policies and compliance

Design and development

Approach
Technical requirements
Architecture
Gathering of training data
Code for data processing
Model creation
Risk treatment

Verification and validation

Verification of data processing
System verification
Risk monitoring and review



Section 3
Deployment

Operation and monitoring

Operating data input
Model execution
Model updates
Risk management

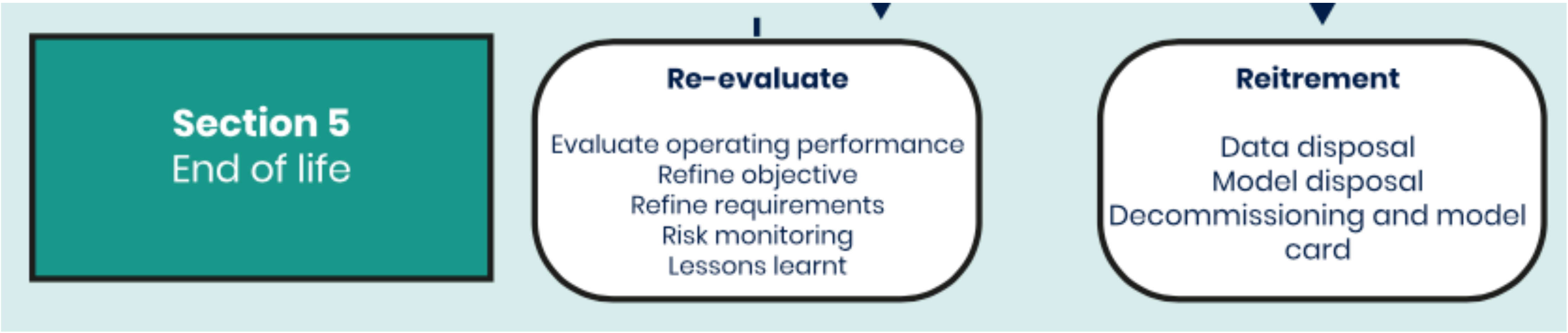
Deployment

Runtime deployment
Model deployment
Risk treatment

Section 4
Continual / online learning

Continuous validation

Validation data processing
System validation
Risk management
Continuous improvement



Section 5
End of life

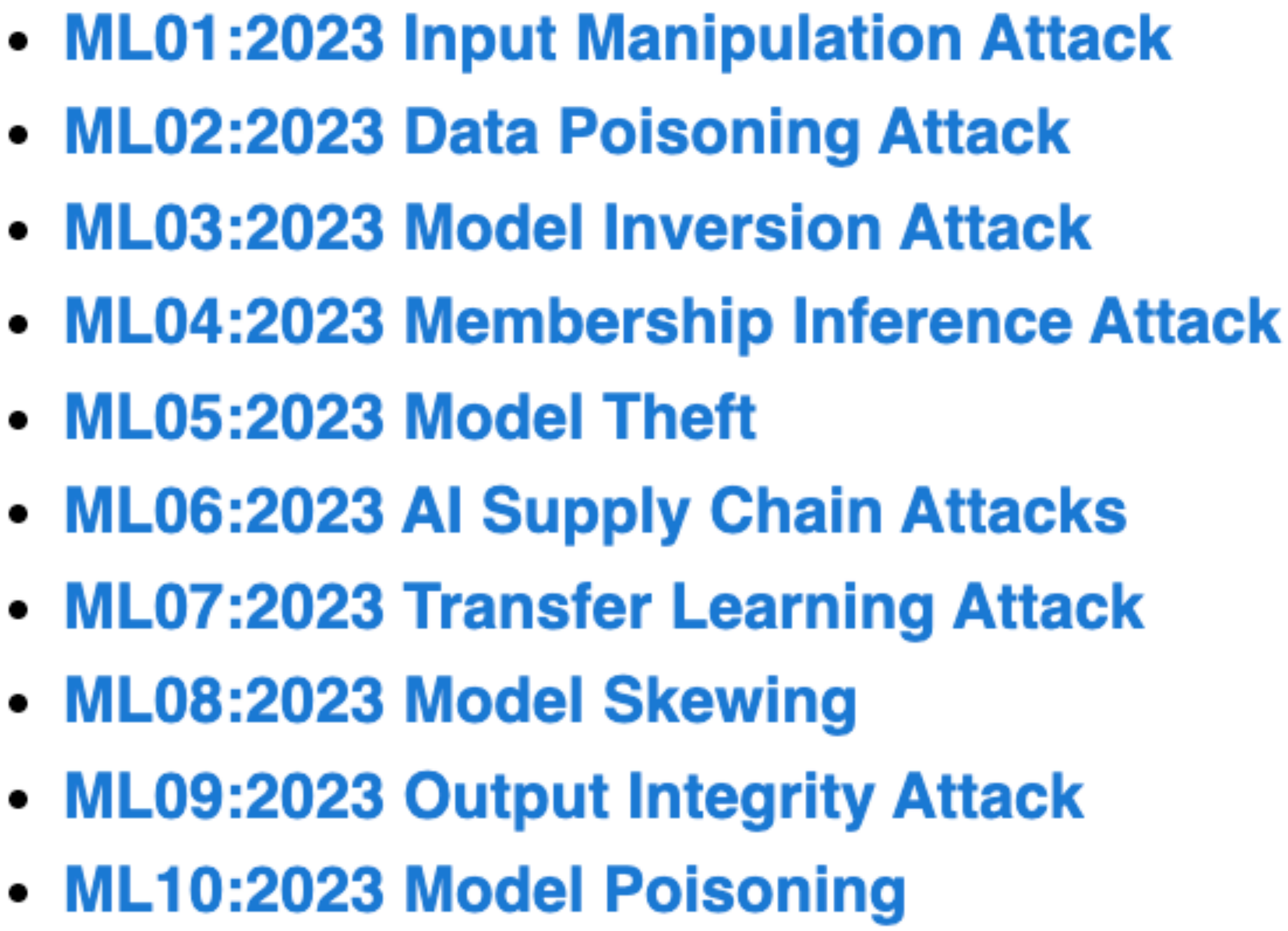
Re-evaluate

Evaluate operating performance
Refine objective
Refine requirements
Risk monitoring
Lessons learnt

Reitrement

Data disposal
Model disposal
Decommissioning and model card

OWASP Top Ten Machine Learning Risks



- ML01:2023 Input Manipulation Attack
- ML02:2023 Data Poisoning Attack
- ML03:2023 Model Inversion Attack
- ML04:2023 Membership Inference Attack
- ML05:2023 Model Theft
- ML06:2023 AI Supply Chain Attacks
- ML07:2023 Transfer Learning Attack
- ML08:2023 Model Skewing
- ML09:2023 Output Integrity Attack
- ML10:2023 Model Poisoning

- https://owasp.org/www-project-machine-learning-security-top-10/

- **ML01:2023 Input Manipulation Attack**
  - An attacker deliberately alters input data to mislead the model
  - This attack is also called **evasion**
  - Example: a model is trained to tell cat images from dog images. An attacker modifies a cat image so it is misclassified as a dog.
- **ML02:2023 Data Poisoning Attack**
  - An attacker manipulates the training data to cause the model to behave in an undesirable way
- **ML03:2023 Model Inversion Attack**
  - An attacker reverse-engineers the model to extract information from it
  - Example: a model is trained to recognize faces. An attacker inputs images of individuals into the model and and recovers the personal information of the individuals from the model's predictions, such as their name, address, or social security number.

- **ML04:2023 Membership Inference Attack**
    - An attacker manipulates the model’s training data in order to cause it to behave in a way that exposes sensitive information
    - Example: A malicious attacker trains a machine learning model on a dataset of financial records and uses it to query whether or not a particular individual’s record was included in the training data.
- **ML05:2023 Model Theft**
    - An attacker gains access to the model’s parameters
    - Example: Stealing a machine learning model from a competitor
- **ML06:2023 AI Supply Chain Attacks**
    - An attacker modifies or replaces a machine learning library or model that is used by a system

- **ML07:2023 Transfer Learning Attack**
  - An attacker trains a model on one task and then fine-tunes it on another task to cause it to behave in an undesirable way
  - Example: An attacker trains a machine learning model on a malicious dataset that contains manipulated images of faces. The attacker then transfers the model's knowledge to a target face recognition system. As a result, the face recognition system starts making incorrect predictions, allowing the attacker to bypass the security and gain access to sensitive information.
- **ML08:2023 Model Skewing**
  - An attacker manipulates the distribution of the training data to cause the model to behave in an undesirable way.
  - Example: The attacker provides fake feedback data to a loan-approving machine learning system. As a result, the model's predictions are skewed, and the attacker's chances of getting a loan approved are significantly increased.

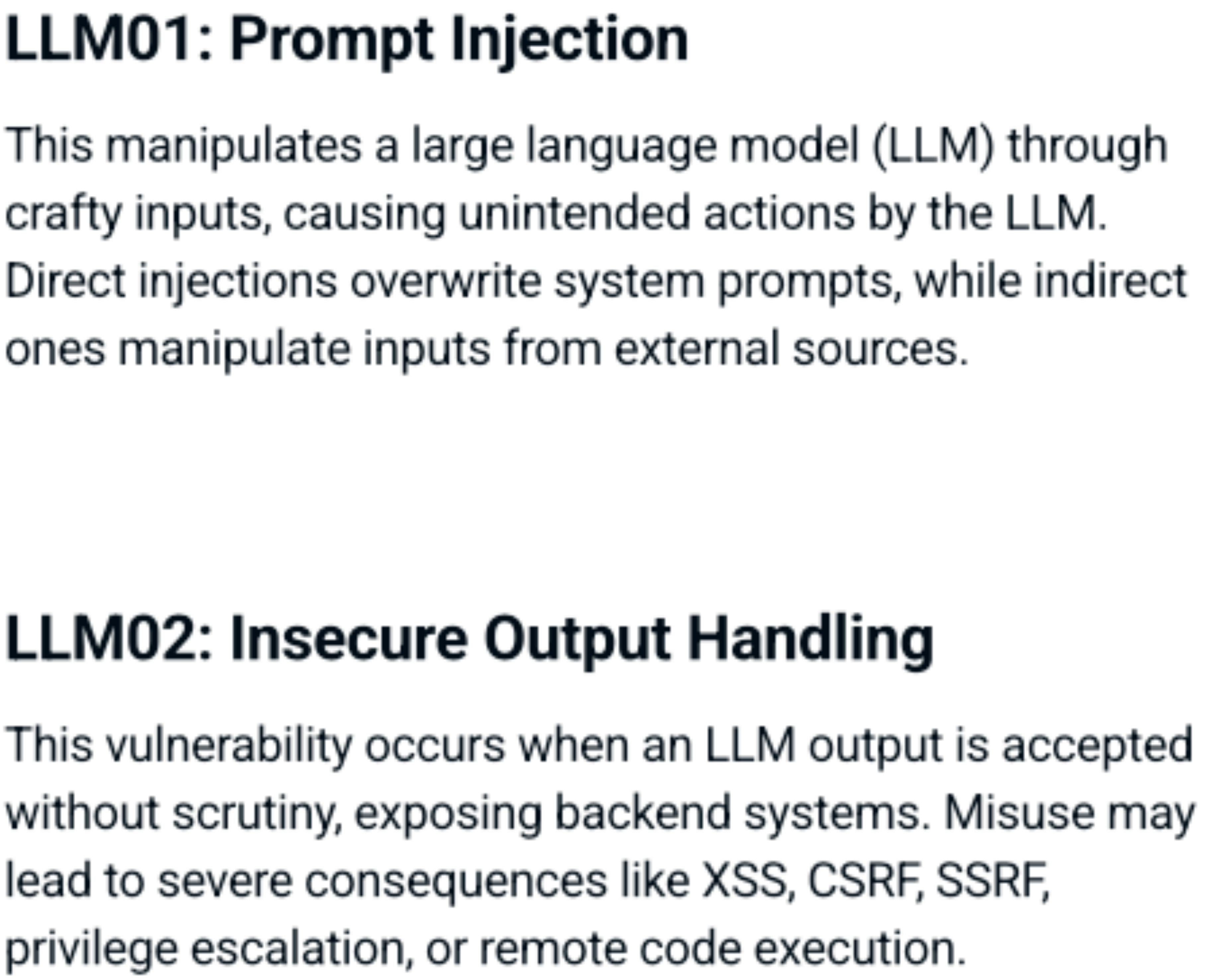
- **ML09:2023 Output Integrity Attack**
    - An attacker aims to modify or manipulate the output of a machine learning model in order to change its behavior or cause harm to the system it is used in.
    - Example: An attacker has gained access to the output of a machine learning model that is being used to diagnose diseases in a hospital. The attacker modifies the output of the model, making it provide incorrect diagnoses for patients.
- **ML10:2023 Neural Net Reprogramming**
    - An attacker manipulates the model's parameters to cause it to behave in an undesirable way.
    - Example: A bank is using a machine learning model to identify handwritten characters on cheques. An attacker manipulates the parameters of the model by altering the images in the training dataset or directly modifying the parameters in the model. This can result in the model misidentifying characters, leading to incorrect amounts being processed.

OWASP Top 10 for LLM Applications

VERSION 1.1

Published: *October 16, 2023*

https://owasp.org/www-project-top-10-for-large-language-model-applications/



LLM01: Prompt Injection

This manipulates a large language model (LLM) through crafty inputs, causing unintended actions by the LLM. Direct injections overwrite system prompts, while indirect ones manipulate inputs from external sources.

LLM02: Insecure Output Handling

This vulnerability occurs when an LLM output is accepted without scrutiny, exposing backend systems. Misuse may lead to severe consequences like XSS, CSRF, SSRF, privilege escalation, or remote code execution.

LLM03: Training Data Poisoning

This occurs when LLM training data is tampered, introducing vulnerabilities or biases that compromise security, effectiveness, or ethical behavior. Sources include Common Crawl, WebText, OpenWebText, & books.

LLM04: Model Denial of Service

Attackers cause resource-heavy operations on LLMs, leading to service degradation or high costs. The vulnerability is magnified due to the resource-intensive nature of LLMs and unpredictability of user inputs.

LLM05: Supply Chain Vulnerabilities

LLM application lifecycle can be compromised by vulnerable components or services, leading to security attacks. Using third-party datasets, pre- trained models, and plugins can add vulnerabilities.

LLM06: Sensitive Information Disclosure

LLMs may inadvertently reveal confidential data in their responses, leading to unauthorized data access, privacy violations, and security breaches. It’s crucial to implement data sanitization and strict user policies to mitigate this.

LLM07: Insecure Plugin Design

LLM plugins can have insecure inputs and insufficient access control. This lack of application control makes them easier to exploit and can result in consequences like remote code execution.

LLM08: Excessive Agency

LLM-based systems may undertake actions leading to unintended consequences. The issue arises from excessive functionality, permissions, or autonomy granted to the LLM-based systems.

LLM09: Overreliance

Systems or people overly depending on LLMs without oversight may face misinformation, miscommunication, legal issues, and security vulnerabilities due to incorrect or inappropriate content generated by LLMs.

LLM10: Model Theft

This involves unauthorized access, copying, or exfiltration of proprietary LLM models. The impact includes economic losses, compromised competitive advantage, and potential access to sensitive information.

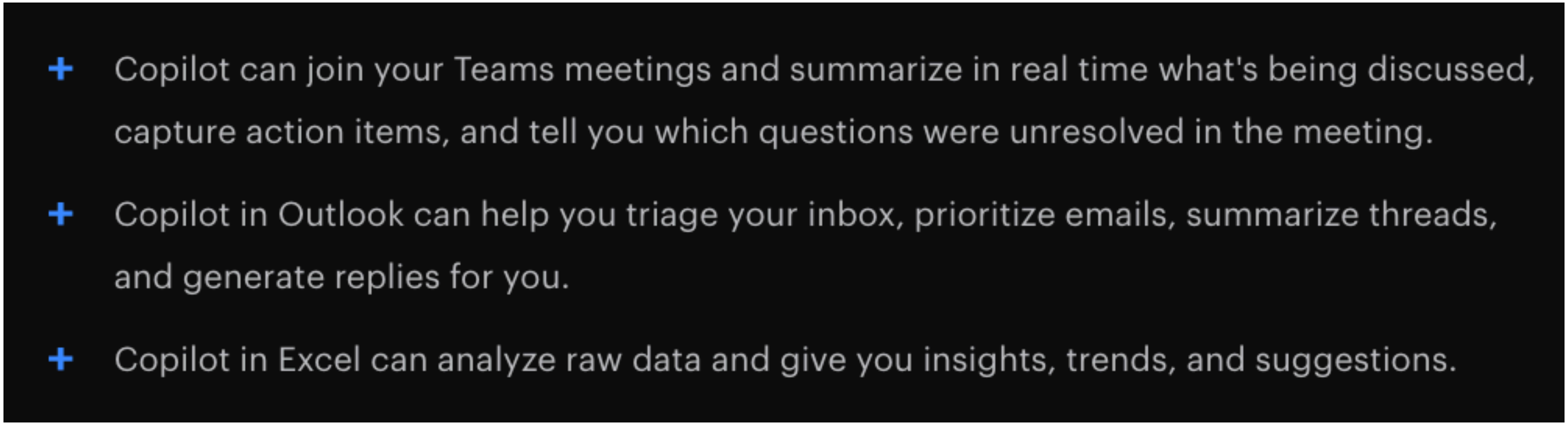
Copilot Security: Ensuring a Secure Microsoft Copilot Rollout

This article describes how Microsoft 365 Copilot's security model works and the risks that must be mitigated to ensure a safe rollout.

Rob Sobers | 5 min read | Last updated April 11, 2024

VARONIS

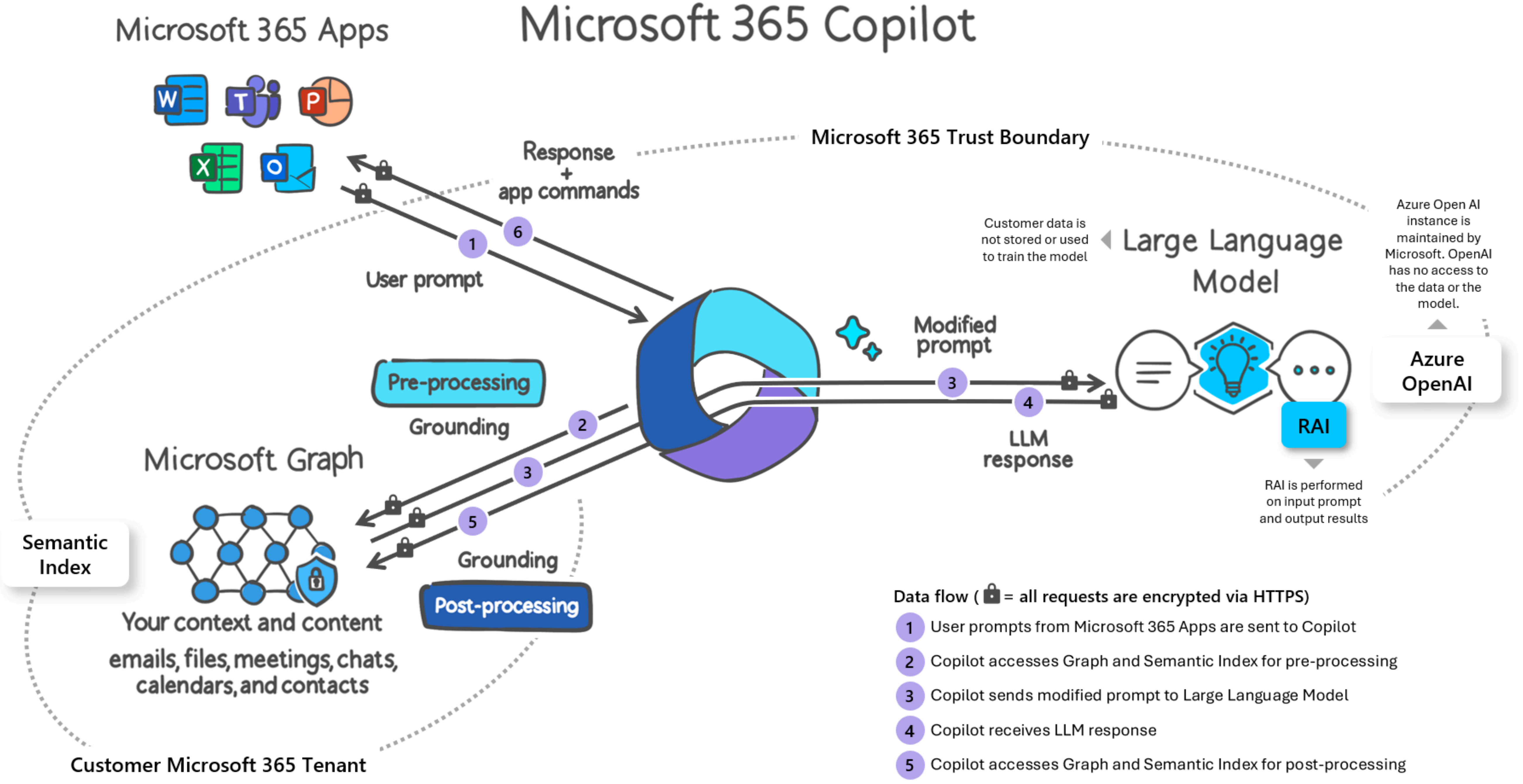
Microsoft 365 Copilot Use Cases



- Copilot can join your Teams meetings and summarize in real time what's being discussed, capture action items, and tell you which questions were unresolved in the meeting.
- Copilot in Outlook can help you triage your inbox, prioritize emails, summarize threads, and generate replies for you.
- Copilot in Excel can analyze raw data and give you insights, trends, and suggestions.

- Writes documents for you
  - Based on data found in your Email, documents, spreadsheets, and other files you have access to
  - In the Microsoft365 cloud
  - **Based on your Microsoft365 permissions**

Microsoft 365 Copilot



Microsoft 365 Apps

Microsoft 365 Trust Boundary

Response + app commands

6

1

User prompt

Customer data is not stored or used to train the model

Large Language Model

Azure Open AI instance is maintained by Microsoft. OpenAI has no access to the data or the model.

Modified prompt

3

4

LLM response

Azure OpenAI

RAI

RAI is performed on input prompt and output results

Pre-processing

Grounding

2

3

5

Grounding

Post-processing

Microsoft Graph

Semantic Index

Your context and content

emails, files, meetings, chats, calendars, and contacts

Customer Microsoft 365 Tenant

Data flow (🔒 = all requests are encrypted via HTTPS)

1 User prompts from Microsoft 365 Apps are sent to Copilot

2 Copilot accesses Graph and Semantic Index for pre-processing

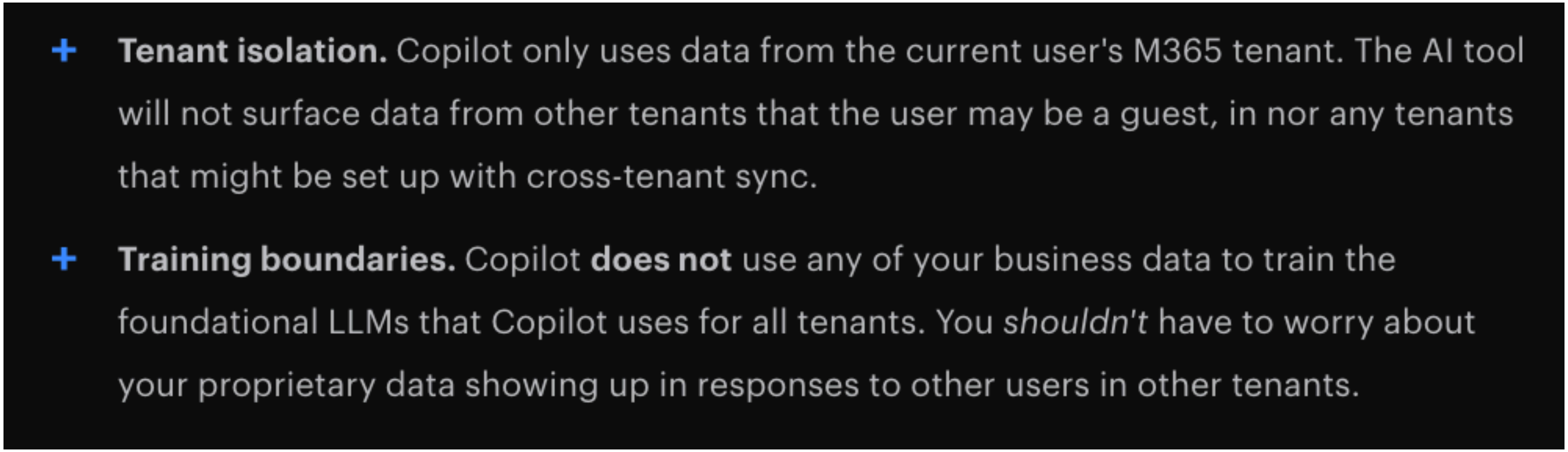
3 Copilot sends modified prompt to Large Language Model

4 Copilot receives LLM response

5 Copilot accesses Graph and Semantic Index for post-processing

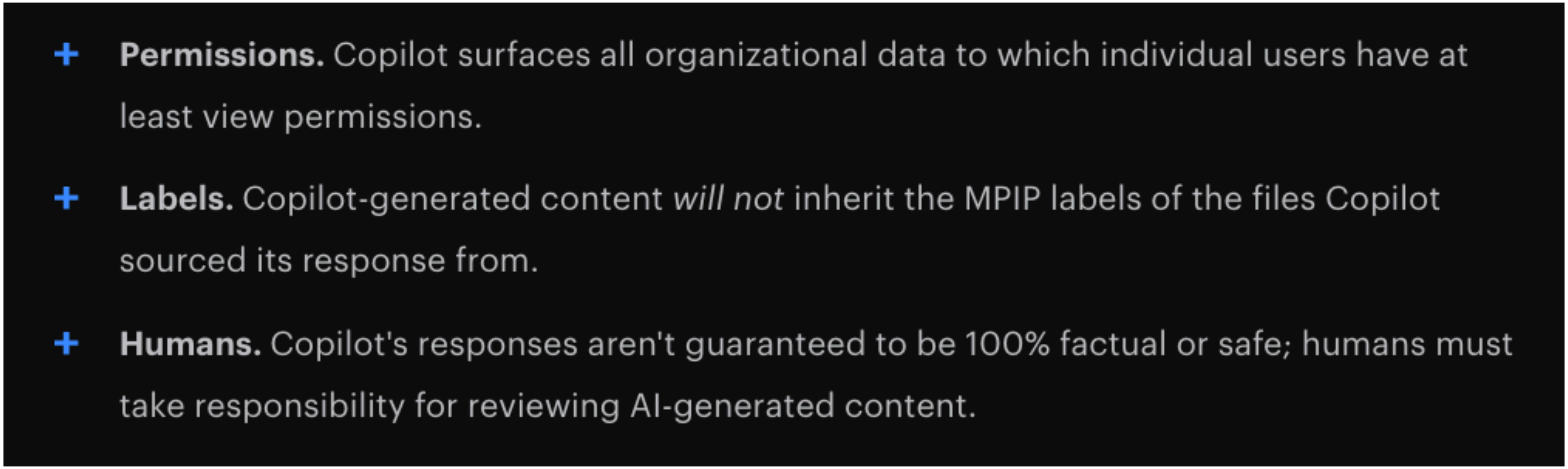
6 Copilot sends the response, and app command back to Microsoft 365 Apps

What Microsoft Handles for You



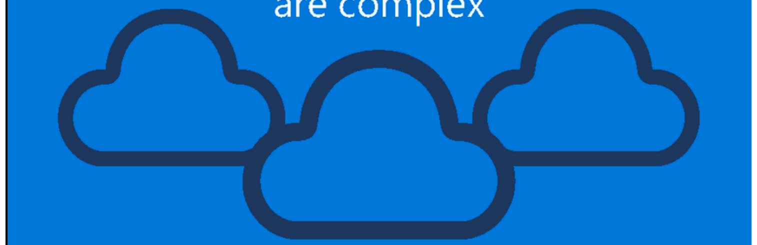
- **Tenant isolation.** Copilot only uses data from the current user's M365 tenant. The AI tool will not surface data from other tenants that the user may be a guest, in nor any tenants that might be set up with cross-tenant sync.
- **Training boundaries.** Copilot **does not** use any of your business data to train the foundational LLMs that Copilot uses for all tenants. You *shouldn't* have to worry about your proprietary data showing up in responses to other users in other tenants.

What You Need to Manage



- **Permissions.** Copilot surfaces all organizational data to which individual users have at least view permissions.
- **Labels.** Copilot-generated content *will not* inherit the MPIP labels of the files Copilot sourced its response from.
- **Humans.** Copilot's responses aren't guaranteed to be 100% factual or safe; humans must take responsibility for reviewing AI-generated content.

Multicloud environments are complex



40,000+

permissions to manage

>50%

are **high-risk**

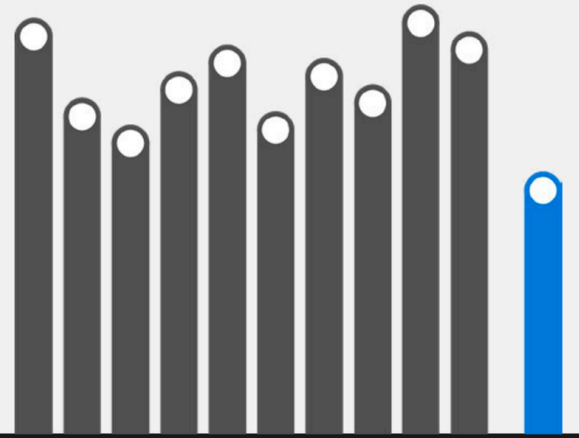
After analyzing over

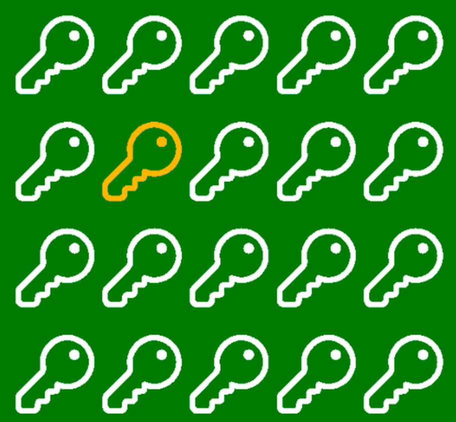
500 risk assessments,

we found that **most identities are greatly over-permissioned**, putting organizations' critical environments at risk for accidental or malicious permission misuse

Workload identities accessing cloud environments are increasing, now outnumbering human identities

10:1





1%

of permissions granted are actually used

>50%

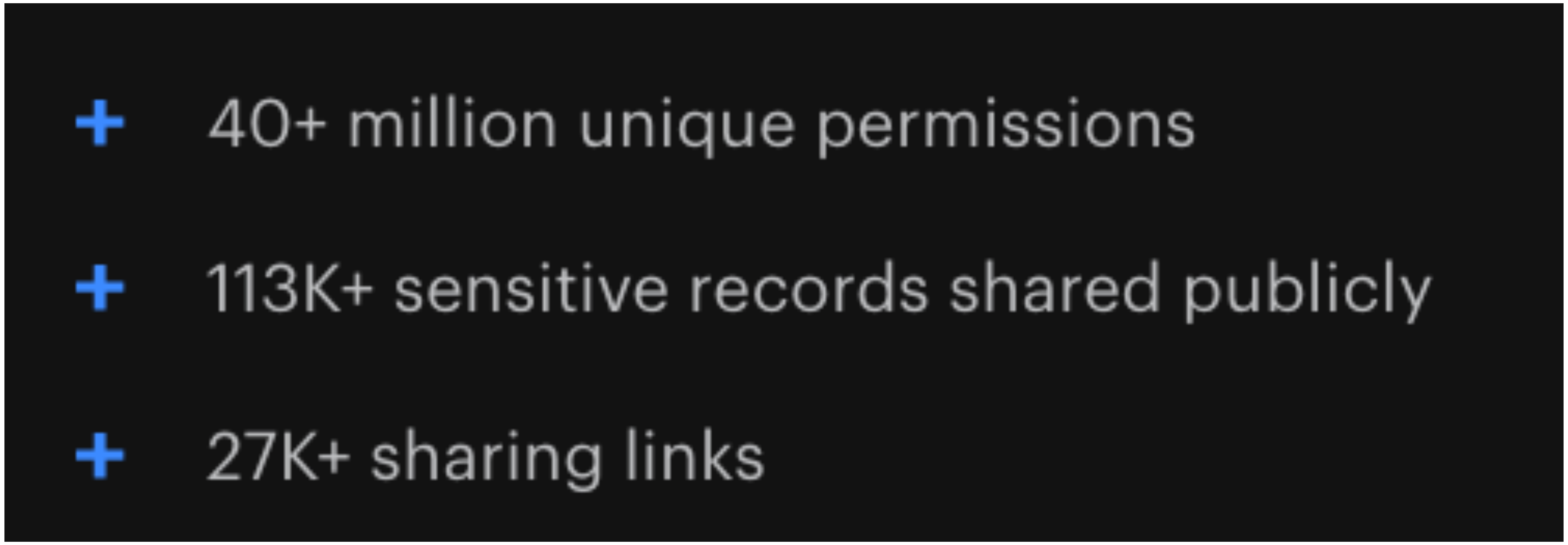
of identities are super admins, meaning they have **access to all permissions and resources**

Microsoft Security

Learn how to implement least privilege and reduce permission risks across multicloud at

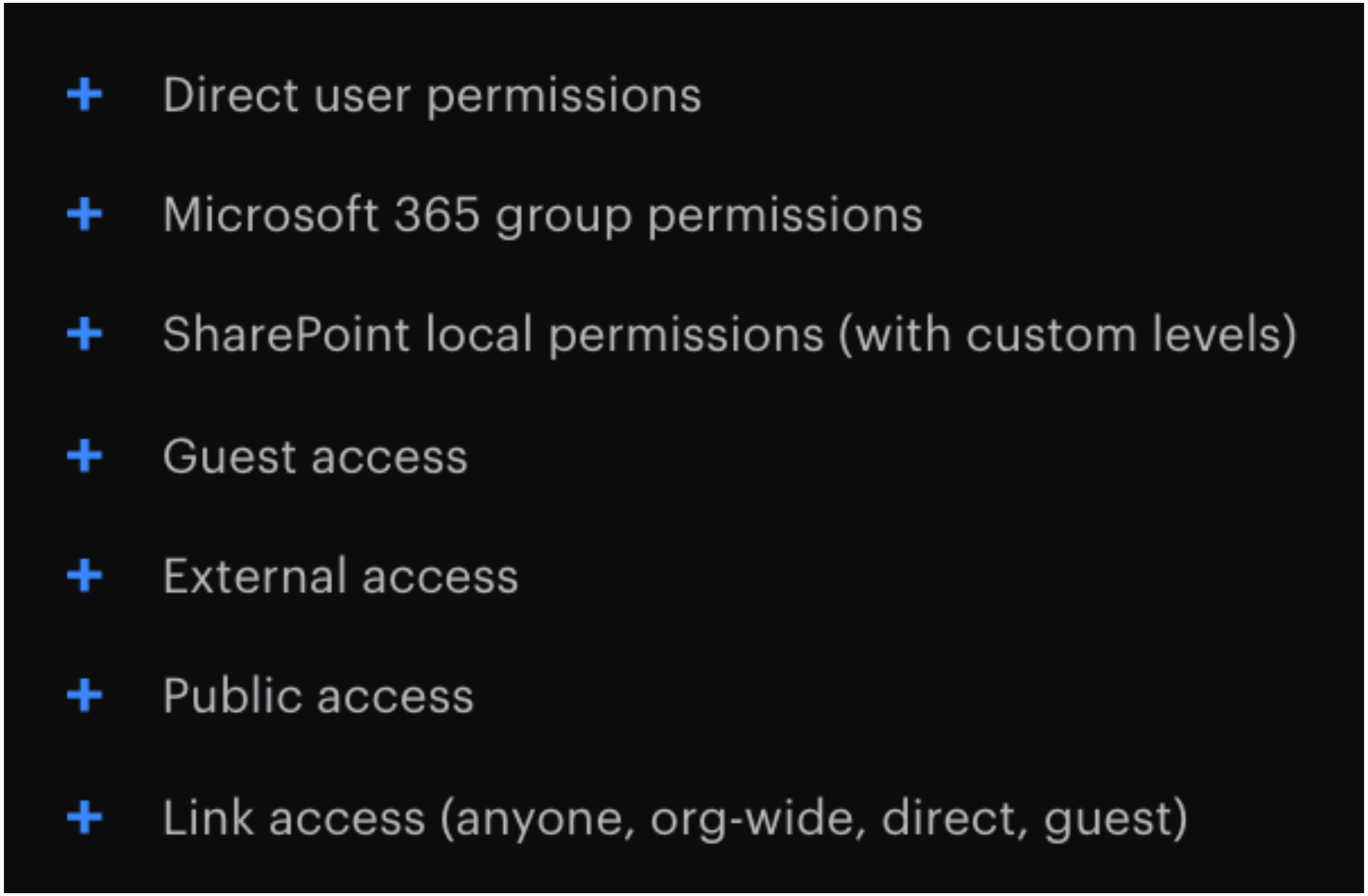
aka.ms/PermissionsManagement

The Average M365 Tenant has

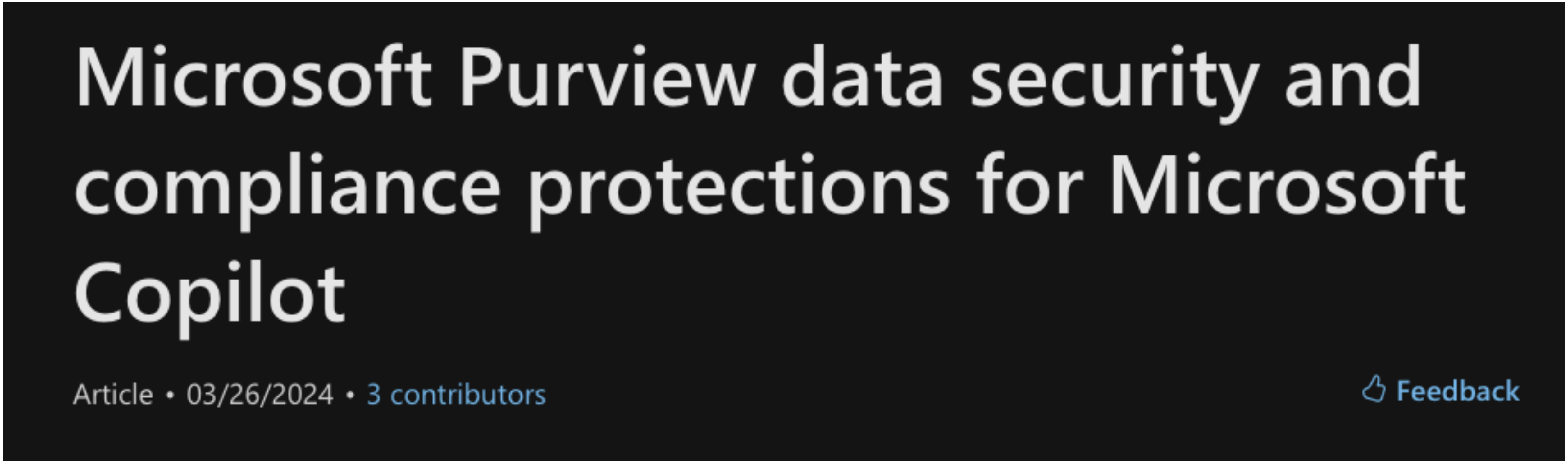


- 40+ million unique permissions
- 113K+ sensitive records shared publicly
- 27K+ sharing links

Why Does This Happen?



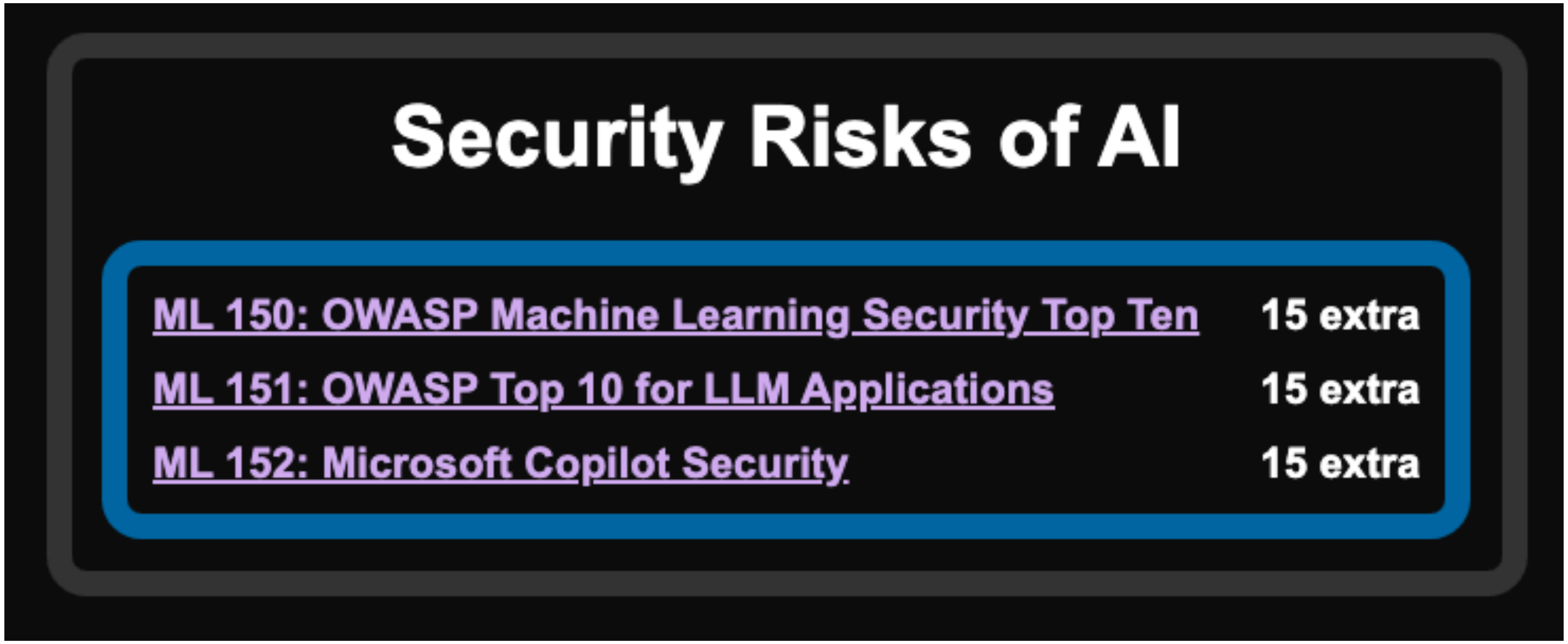
- Direct user permissions
- Microsoft 365 group permissions
- SharePoint local permissions (with custom levels)
- Guest access
- External access
- Public access
- Link access (anyone, org-wide, direct, guest)



Microsoft Purview data security and compliance protections for Microsoft Copilot

Article • 03/26/2024 • 3 contributors

- But you must enable sensitivity labels
  - For SharePoint and OneDrive
- If humans fail to apply and update labels, the system fails



Security Risks of AI

ML 150: OWASP Machine Learning Security Top Ten	15 extra
ML 151: OWASP Top 10 for LLM Applications	15 extra
ML 152: Microsoft Copilot Security	15 extra