Assentian Limited

PRIVACY ENHANCING TECHNOLOGIES AS AN ENABLER OF TRUSTED AND RESPONSIBLE AI

Dr Ilesh Dattani CEng CISA
Mr Joshua Priestley

ISACA Hong Kong Workshop

ABOUT

WHO WE ARE

A global Cyber Security and Blockchain Innovation Hub working with clients and partners to address current business challenges by exploring the potential of emerging technologies.

- Over 25 Years of Experince in working with emerging technologies like AI, Machine Learning, Internet of Things, Blockchain and others
- Support with regulatory compliance around Cyber Security and Data Protection including ISO 27001, PCI DSS, GDPR, LGPD and others
- Cerified Information Security Auditors, ISO 27001 and CREST Accredited
- Approved UK Crown Commercial Services Suppliers

JPMorganChase

EUROPEAN CENTRAL BANK
EUROSYSTEM

U.S. DEPARTMENT OF HOMELAND SECURITY

Westpac

vodafone

Telstra

CREST

CISA Certified Information Systems Auditor. An ISACA® Certification

bsi ISO/IEC 27001 Information Security Management CERTIFIED

bsi ISO 9001 Quality Management Systems CERTIFIED

Crown Commercial Service

Assentian Limited

Agenda

- Welcome and Introductions
- Introduction to Responsible and Trusted AI
- Deep Dive into Privacy Enhancing Technologies (PETs)
- PETs Across the AI Lifecycle
- Challenges, Governance, and Regulatory Landscape
- Future Trends
- Wrap-Up

Housekeeping

- Optional – Practical Demos
- Github: https://github.com/idattani/Hong-Kong-Nov-25
- Google Colab: https://colab.google/

Responsible AI Video

What is responsible AI?

Responsible artificial intelligence (AI) is a set of principles that help guide the design, development, deployment and use of AI—building trust in AI solutions that have the potential to empower organizations and their stakeholders. Responsible AI involves the consideration of a broader societal impact of AI systems and the measures required to align these technologies with stakeholder values, legal standards and ethical principles. Responsible AI aims to embed such ethical principles into AI applications and workflows to mitigate risks and negative outcomes associated with the use of AI, while maximizing positive outcomes.

What is responsible AI?

- **Transparency** - ensuring systems are open to scrutiny, with meaningful information provided to relevant individuals across their lifecycle.
- **Accountability** - ensuring systems have effective governance and oversight mechanisms, with clear lines of appropriate responsibility across their lifecycle.
- **Human-centred Value** - ensuring systems have a clear purpose and benefit to individuals, and are designed with humans in mind.
- **Fairness** - ensuring systems are designed and deployed against an appropriate definition of fairness, and monitored for fair use and outcomes.
- **Privacy** - ensuring systems are privacy-preserving, and the rights of individuals around their personal data are respected
- **Safety** - ensuring systems behave reliably as intended, and their use does not inflict undue physical or mental harms.
- **Security** - ensuring systems are measurably secure and resistant to being compromised by unauthorised parties.
- **Societal Wellbeing** - ensuring systems support beneficial outcomes for societies and the planet.

What is responsible AI?

Underlying the Fundamentals are the Conditions. These are the technical, organisational and environmental factors that must be satisfied in order for the Fundamentals to be met. Located on the inner ring of the Model, they are:

- **Meaningful Engagement** - engaging effectively with experts, stakeholders, and the general public, using these insights to inform the system in question.
- **Robust Technical Design** - ensuring that the functional (how a program will behave to outside agents) and technical (how that functionality is implemented in code) design of a system is robust.
- **Appropriate & Available Data** - ensuring a system has access to the right data needed to achieve its desired outcomes and effectively monitor performance.
- **Clear Boundaries** - ensuring there are clear boundaries on a system's intended use, and clear understanding of the consequences of exceeding them.
- **Available Resources** - ensuring the resources (technical, legal, financial, etc.) needed to effectively build and use a system are provided.
- **Effective Governance** - ensuring that the right processes and policies are in place to guide the development and operation of a system, and ensure its adherence to the project's goals, standards and regulations, providing recourse where necessary.

The Pillars of Trust

Explainability

Prediction accuracy

Accuracy is a key component of how successful the use of AI is in everyday operation. By running simulations and comparing AI output to the results in the training data set, the prediction accuracy can be determined. The most popular technique used for this is Local Interpretable Model-Agnostic Explanations (LIME), which explain the prediction of classifiers by the machine learning algorithm.

Traceability

Traceability is a property of AI that signifies whether it allows users to track its predictions and processes. It involves the documentation of data and how it is processed by models. Traceability is another key technique for achieving explainability, and is accomplished, for example, by limiting the way decisions can be made and setting up a narrower scope for machine learning rules and features.

Decision understanding

This is the human factor. Practitioners need to be able to understand how and why AI derives conclusions. This is accomplished through continuous education.

JPMorganChase

AI MODEL EVALUATOR

CREST

CISA Certified Information Systems Auditor. An ISACA® Certification

bsi ISO/IEC 27001 Information Security Management CERTIFIED

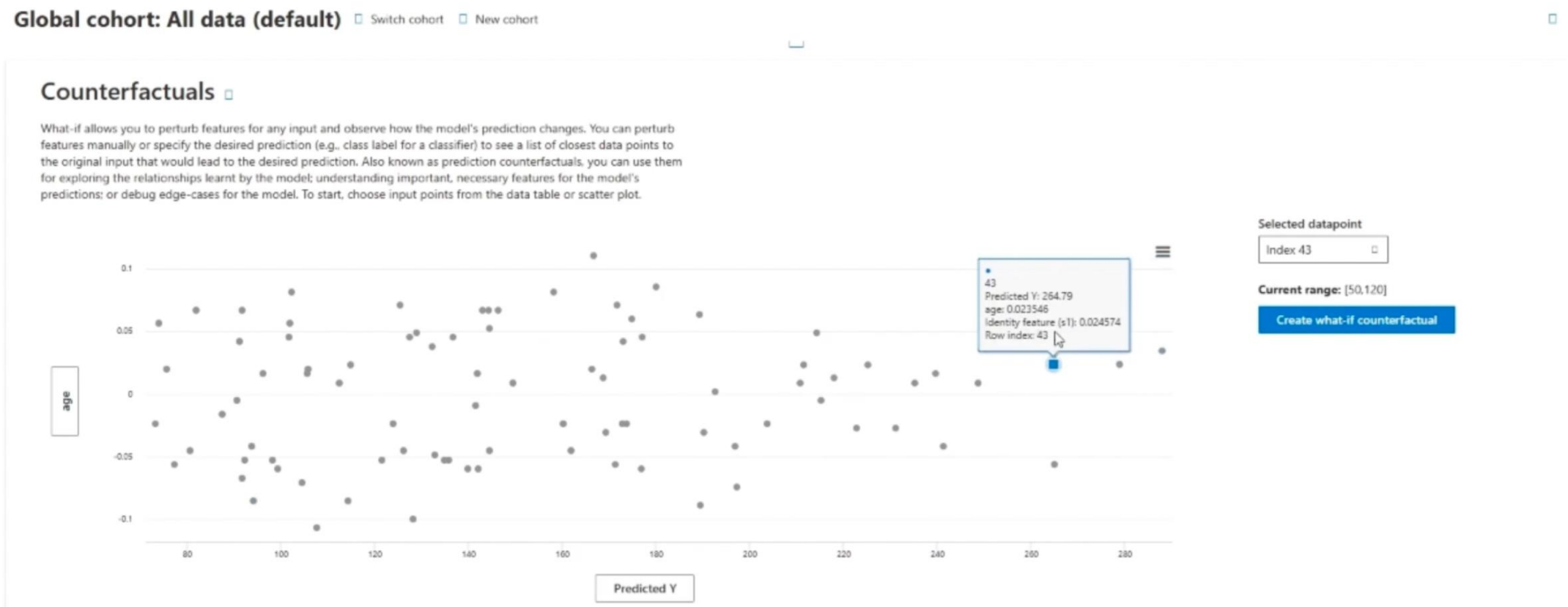
bsi ISO 9001 Quality Management Systems CERTIFIED

Crown Commercial Service

Assentian Limited

The Pillars of Trust

AI MODEL EVALUATOR



Certified Information Systems Auditor. An ISACA® Certification

ISO/IEC 27001 Information Security Management CERTIFIED

ISO 9001 Quality Management Systems CERTIFIED

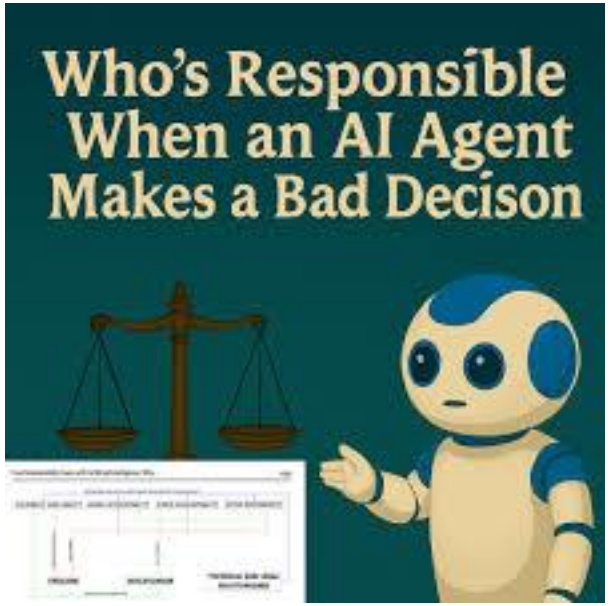
Crown Commercial Service

Assentian Limited

The Pillars of Trust

Fairness

Machine learning models are increasingly used to inform high stakes decision-making that relates to people. Although machine learning, by its very nature, is a form of statistical discrimination, the discrimination becomes objectionable when it places privileged groups at systematic advantage and certain unprivileged groups at systematic disadvantage, potentially causing varied harms. Biases in training data, due to either prejudice in labels or under-/over-sampling, yields models with unwanted bias.



- **Diverse and representative data**
- **Bias-aware algorithms**
- **Bias mitigation techniques**

The Pillars of Trust

Privacy

Many regulatory frameworks, including GDPR, mandate that organizations abide by certain privacy principles when processing personal information. A malicious third party with access to a trained ML model, even without access to the training data itself, can still reveal sensitive personal information about the people whose data was used to train the model. It is crucial to be able to protect AI models that may contain personal information, and control what data goes into the model in the first place.

The Pillars of Trust

Requirements
Data Minimization: **Goal:** Minimize the collection, storage, and processing of personal data to the extent necessary for achieving the intended purpose. **Usage:** AI systems should only collect and retain the minimum amount of data required to fulfil their functions, reducing the risk of privacy breaches and limiting exposure to potential misuse.
Purpose Limitation: **Goal:** Limit the use of personal data to specific, legitimate purposes disclosed to the data subjects at the time of collection. **Usage:** AI systems should process personal data only for the purposes for which it was collected or consented to by the individuals, ensuring transparency and accountability in data processing activities.
Data Accuracy and Integrity: **Goal:** Ensure the accuracy, completeness, and integrity of personal data processed by AI systems. **Usage:** AI systems should employ measures to maintain data accuracy and integrity, such as data validation, verification, and correction mechanisms, to prevent inaccuracies and errors that could impact individuals' rights and interests.
User Consent and Control: **Goal:** Obtain informed consent from individuals for the collection, use, and sharing of their personal data and provide them with control over their data. **Usage:** AI systems should incorporate mechanisms for obtaining explicit consent from users for data processing activities and offer options for users to manage their privacy preferences, access, rectify, or delete their data.
Transparency and Accountability: **Goal:** Ensure transparency in AI system operations, decision-making processes, and data processing activities, and establish accountability for compliance with privacy laws and regulations. **Usage:** AI systems should provide users with clear and accessible information about how their data is collected, used, and shared, as well as the purposes and implications of AI-driven decisions. Organizations should implement measures to monitor, audit, and demonstrate compliance with privacy requirements, fostering trust and accountability.
Security Safeguards: **Goal:** Implement robust security measures to protect personal data against unauthorized access, disclosure, alteration, or destruction. **Usage:** AI systems should incorporate security safeguards, such as encryption, access controls, authentication mechanisms, and regular security assessments, to safeguard personal data from cyber threats, data breaches, and unauthorized use.

The Pillars of Trust

Requirements

Data Security:

- Data Encryption: Encrypt sensitive data at rest and in transit to prevent unauthorized access.
- Access Controls: Implement access controls to restrict data access to authorized users and roles.
- Data Minimization: Minimize the collection and retention of sensitive data to reduce the attack surface.
- Data Integrity: Ensure the integrity of data by implementing measures to detect and prevent data tampering.

Model Security:

- Model Encryption: Encrypt trained models to protect intellectual property and prevent model theft.
- Model Versioning: Maintain version control of models to track changes and ensure reproducibility.
- Model Validation: Validate models for accuracy, fairness, and robustness to mitigate risks of biased or inaccurate predictions.
- Model Auditing: Conduct regular audits of models to detect and mitigate vulnerabilities and ensure compliance with security standards.

Access Control and Authentication:

- User Authentication: Implement strong authentication mechanisms, such as multi-factor authentication (MFA), to verify the identities of users.
- Role-based Access Control (RBAC): Assign roles and permissions to users based on their responsibilities and privileges.
- API Security: Secure APIs used for accessing and interacting with AI systems through authentication, authorization, and encryption.
- Session Management: Manage user sessions securely to prevent session hijacking and unauthorized access.

Network Security:

- Firewalls and Intrusion Detection Systems (IDS): Deploy firewalls and IDS to monitor and control network traffic and detect suspicious activities.
- Secure Communication Protocols: Use secure communication protocols (e.g., HTTPS, SSL/TLS) to protect data transmission between AI systems and other components.
- Network Segmentation: Segment networks to isolate AI systems from other systems and restrict communication to authorized entities.

Group Exercise:
Identifying Privacy Challenges in AI Systems

Identifying Privacy Challenges in AI Systems

- **AI Use Cases**
  - Healthcare diagnostics using patient data
  - Financial fraud detection across institutions
  - Social media content moderation
  - Smart city traffic management with sensor data
  - Employee hiring and performance evaluation AI
    - OR Your own use case

Identifying Privacy Challenges in AI Systems

- Get into Groups
- Phase 1 (10 Minutes)
  - Define the AI Use Case Context:
  - Choose one of the given AI use cases (healthcare diagnostics, financial fraud detection, social media moderation, smart city traffic management, employee hiring/performance OR your own use case).
  - Describe the data involved and AI functionalities at a high level.
- Phase 2 (15 Minutes)
  - Identify Privacy Concerns:
  - List potential privacy risks (e.g., unauthorized data access, re-identification, surveillance, bias in decisions).
  - Consider specific types of personal data processed and sensitivity levels.
  - Think about unintended privacy harms arising from AI use.

Identifying Privacy Challenges in AI Systems

- Phase 3: (15 Minutes)
  - Determine Relevant Regulatory Requirements:
    - Identify applicable data protection and AI governance regulations (e.g., GDPR, HIPAA, CCPA, Digital Services Act, local laws).
    - Highlight obligations such as data subject rights, consent, data minimization, purpose limitation, fairness, transparency, and accountability.
    - Consider sector-specific or cross-jurisdictional rules.
  - Propose Mitigations and Solutions:
    - Could be specific technologies
    - Organizational and procedural controls such as transparency policies, consent management, human oversight, and auditability.

Identifying Privacy Challenges in AI Systems

- Phase 4 – all together (15 Minutes)
  - Present and Justify the Approach:
    - Explain how the chosen mitigations address the identified privacy concerns and meet regulatory requirements.
    - Consider practicality, effectiveness, and scalability of the proposed solutions.
    - Reflect on any residual risks and how ongoing monitoring could be integrated.

Data Anonymization

What Is Data Anonymization

Data anonymization is the process of protecting private or sensitive information by erasing or encrypting identifiers that connect an individual to stored data. For example, you can run Personally Identifiable Information (PII) such as names, social security numbers, and addresses through a data anonymization process that retains the data but keeps the source anonymous.

Data Anonymization Techniques

- **Data masking**—hiding data with altered values. You can create a mirror version of a database and apply modification techniques such as character shuffling, encryption, and word or character substitution. For example, you can replace a value character with a symbol such as “*” or “x”. Data masking makes reverse engineering or detection impossible.
- **Pseudonymization**—a data management and de-identification method that replaces private identifiers with fake identifiers or pseudonyms, for example replacing the identifier “John Smith” with “Mark Spencer”. Pseudonymization preserves statistical accuracy and data integrity, allowing the modified data to be used for training, development, testing, and analytics while protecting data privacy.

Data Anonymization Techniques

- **Generalization**—deliberately removes some of the data to make it less identifiable. Data can be modified into a set of ranges or a broad area with appropriate boundaries. You can remove the house number in an address, but make sure you don't remove the road name. The purpose is to eliminate some of the identifiers while retaining a measure of data accuracy.
- **Data swapping**—also known as shuffling and permutation, a technique used to rearrange the dataset attribute values so they don't correspond with the original records. Swapping attributes (columns) that contain identifiers values such as date of birth, for example, may have more impact on anonymization than membership type values.

Data Anonymization Techniques

- **Data perturbation**—modifies the original dataset slightly by applying techniques that round numbers and add random noise. The range of values needs to be in proportion to the perturbation. A small base may lead to weak anonymization while a large base can reduce the utility of the dataset. For example, you can use a base of 5 for rounding values like age or house number because it's proportional to the original value. You can multiply a house number by 15 and the value may retain its credence. However, using higher bases like 15 can make the age values seem fake.

Data Anonymization Techniques

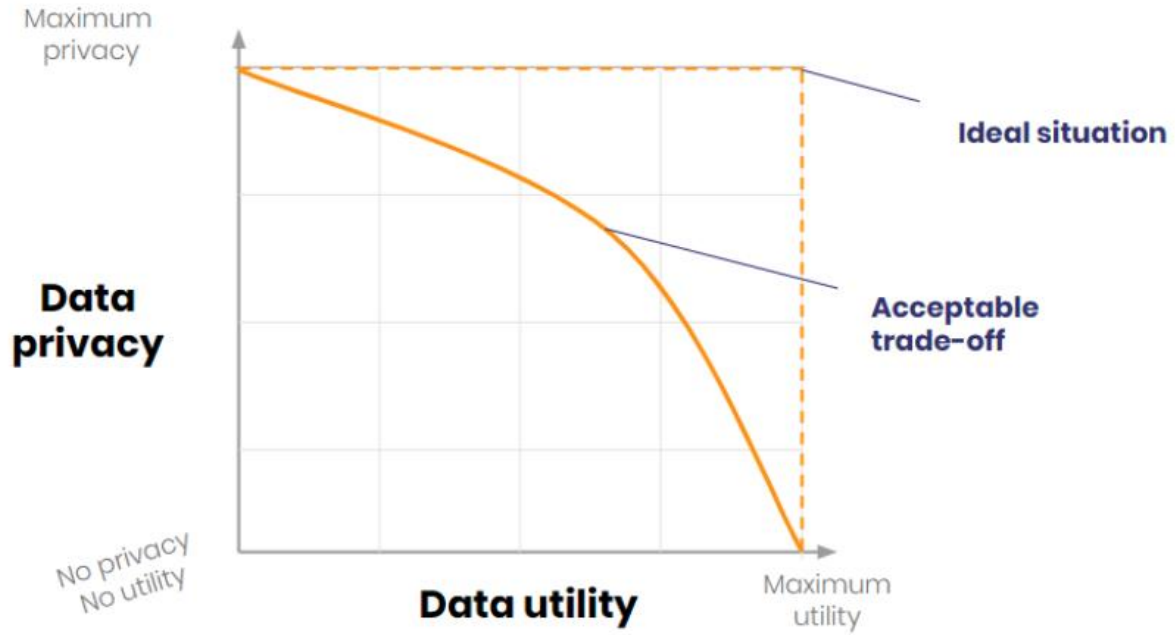
- **K-anonymity**: A method where data is transformed so that each person is indistinguishable from at least *k-1* others in the dataset.

  Imagine you have a dataset that contains the attributes of age, gender, and zip codes for a subset of customers. To make the data K anonymous with a value of K=4, we need to ensure that for every combination of age, gender, and zip code, there are at least four individuals with the same values. That would require generalizing or suppressing some information, such as replacing exact ages with an age range, or replacing the zip code with a larger geographic region.

DATA ANONYMISATION DEMO

Disadvantages of Data Anonymization

The GDPR stipulates that websites must obtain consent from users to collect personal information such as IP addresses, device ID, and cookies. Collecting anonymous data and deleting identifiers from the database limit your ability to derive value and insight from your data. For example, anonymized data cannot be used for marketing efforts, or to personalize the user experience.



The Risk of Re-Identification

One of the most significant concerns with data anonymization is the risk of **re-identification**. Even anonymized data can sometimes be re-identified by cross-referencing it with other data sets. For example, publicly available information such as voter rolls, social media profiles, or online databases can be used to reverse-engineer anonymized data sets.

Massachusetts Group Insurance Commission

- Public release of hospital visit records for every state employee
  - Very sensitive!
- Anonymized data
  - Original: Name, SSN, ZIP code, date of birth, sex, condition
  - Anonymized: ~~Name~~, ~~SSN~~, ZIP code, date of birth, sex, condition
- Latanya Sweeney: bought voter rolls
  - Contain name, address, ZIP code, date of birth, sex

PETS in AI Video

What are privacy-enhancing technologies (PETs)?

These technologies are designed to safeguard data-in-use processes while allowing the system to perform its essential functions. Pets are specifically engineered to accomplish the following

- Execute trusted computation in an untrusted environment.
- Extract insights from private data without disclosing the sensitive contents of the data.
- Enable parties to work together while ensuring that any shared data is used solely for its intended purpose.
- Incorporate quantum-resistant data protections into the system.
- Ensure that sensitive data is not disclosed while accessing shared artificial intelligence (AI) models.
- Enhance the capacity of data proprietors to maintain control over their data throughout its lifecycle.

Homomorphic Encryption: How It Works

What is homomorphic encryption?

Fully homomorphic encryption (FHE) is an innovative technology that can help you achieve zero trust by unlocking the value of data on untrusted domains without needing to decrypt it.

Today's business data is stored across hybrid multicloud environments, exposing it to various security and privacy risks. While encryption provides protection, the sensitive data typically must first be decrypted to access it for computing and business-critical operations.

This opens the door to potential compromise of privacy and confidentiality controls. Until now, those vulnerabilities have been the cost of doing business in the cloud and with third parties.

With fully homomorphic encryption, you can better enforce zero trust because the data is always encrypted and can be shared, even on untrusted domains in the cloud, while remaining unreadable by those doing the computation.

In short, one can now do high-value analytics and data processing, by internal or external parties, without requiring that data to be exposed.

What is homomorphic encryption?

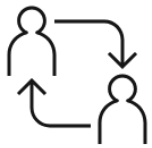
The lattice-based cryptography central to Fully Homomorphic Encryption provides long-term protection and privacy by allowing data to remain encrypted for the duration of its lifecycle, even while computations are being performed. Because the party accessing the data can never “see” it, sensitive material and proprietary algorithms are continuously protected against internal and external threats of data exfiltration.

This protection is extended to quantum computing, as homomorphically encrypted data is proven to be quantum-safe. With traditionally encrypted data, cyber attackers can harvest encrypted data and encryption keys for future use, thus breaking the encryption with the power of quantum computing. However, the homomorphic encryption scheme uses a lattice key approach which prevents this vulnerability, even from quantum computing. With these capabilities, FHE makes accessing and computing data fully secure and private.

Benefits of homomorphic encryption

Gain valuable insights

Generate measurable economic benefits by allowing lines of business and third parties to perform big data analytics on encrypted data while maintaining privacy and compliance controls.



Collaborate confidently on hybrid cloud

Process encrypted data in public and private clouds and third-party environments while maintaining confidentiality controls.

Enable AI, analytics and machine learning (ML)

Use AI and ML to compute upon encrypted data without exposing sensitive information.

CREST

CISA Certified Information Systems Auditor. An ISACA® Certification

bsi ISO/IEC 27001 Information Security Management CERTIFIED

bsi ISO 9001 Quality Management Systems CERTIFIED

Crown Commercial Service

Assentian Limited

What Are the Main Types of Homomorphic Encryption?

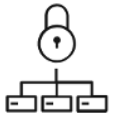
Several homomorphic encryption schemes exist, each with its properties and use cases. The three main types, however, are:

- **Partially Homomorphic Encryption (PHE):** This type allows you to perform only one type of operation (either addition or multiplication) on encrypted data.
- **Somewhat Homomorphic Encryption (SHE):** This type allows you to perform limited addition and multiplication operations on encrypted data. As such, practical limitations apply to the number of operations you can do before the noise in the encrypted data makes further computations unreliable.
- **Fully Homomorphic Encryption (FHE):** This type allows unlimited addition and multiplication operations on encrypted data. It is the most advanced form of homomorphic encryption but is generally more computationally intensive.

Big picture

1. Data owner encrypts and sends sensitive information to a data processor.

2. The data processor receives encrypted information, and can perform computations on the data while it remains encrypted by using a public key.



3. The encrypted results are sent back to the data owner, and are never viewed by the data processor.

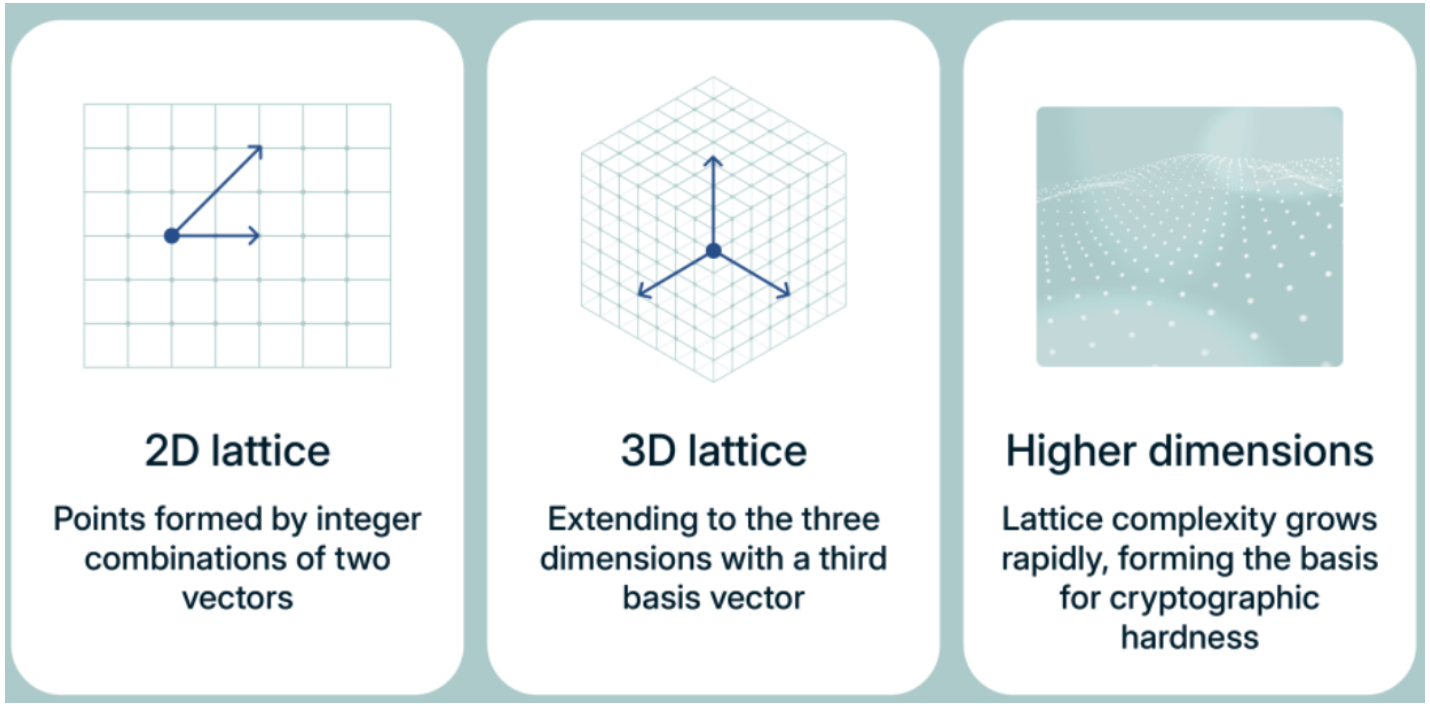
4. The data owner can now decrypt the data and results for further use by using a secret key (KYOK) for technical assurance.

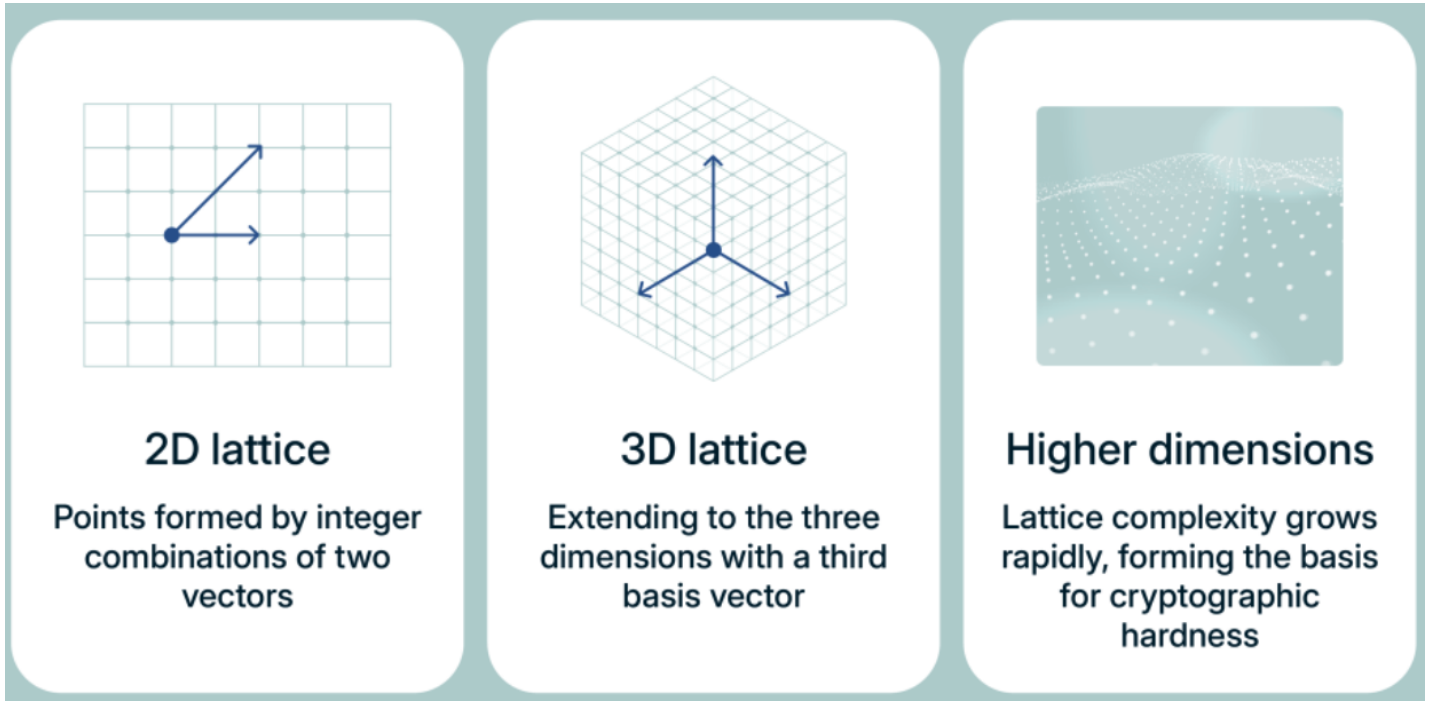
FHE Video

What is a lattice in cryptography?

Imagine looking at a sheet of graph paper with its neat 2D grid pattern. Every intersection follows a set of mathematical rules based on how its lines (vectors) combine.

Lattices extend this concept into hundreds or even thousands of dimensions, creating infinite point sets that follow predictable spacing but become extraordinarily difficult to analyze in higher dimensions.





These high-dimensional lattices are packed with hard math problems. One of the most famous is the **Shortest Vector Problem** (SVP): finding the shortest non-zero vector in a lattice. It sounds simple, but as the number of dimensions explodes, this problem becomes nearly impossible to solve. Think of it like searching for the tiniest puzzle piece in a massive 3D puzzle without knowing what the final picture looks like.

How lattice-based public key encryption works

At its core, lattice encryption hides messages inside math problems that are easy to create but extremely hard to reverse without the correct key.

A common approach is built on the **Learning With Errors** problem: solving slightly "noisy" linear equations. Without the secret key, it's like trying to complete a puzzle with missing pieces. The private key acts as a trapdoor, allowing the intended recipient to filter out noise and recover the original message.

Another well-known method is **NTRU**, which uses polynomial operations instead of matrices. Its public key behaves like a one-way function: simple to compute but very hard to reverse. NTRU is fast, has relatively small key sizes, and has withstood decades of cryptanalysis, making it a strong choice for real-world deployment.

Lattice Based Cryptography

Applications and advantages

- **Post-quantum cryptography:**
  - Lattice-based schemes are a leading candidate for securing communications and data in the quantum era.
- **Efficiency:**
  - Lattice-based schemes can be quite efficient, often using linear operations that can be parallelized.
- **Fully Homomorphic Encryption (FHE):**
  - The study of lattices has also led to advances in areas like FHE, which allows for computations on encrypted data.
- **Digital signatures:**
  - Lattice-based schemes can be used to create digital signatures, such as CRYSTALS-Dilithium, which are also resistant to quantum attacks.

Find out more at
https://www.expressvpn.com/blog/lattice-based-cryptography/

What Are the Applications of Homomorphic Encryption?

- **Securing cloud computing:** Homomorphic encryption lets users store their data in the cloud in encrypted form. A cloud service provider can perform computations on the encrypted data without decrypting it, ensuring the confidentiality of the sensitive information.

- **Preserving privacy during data analysis:** In scenarios where multiple parties want to jointly analyze data without revealing its raw form, homomorphic encryption enables secure and private collaboration. Each party can encrypt its data, share it with others, and perform computations on the other entity's encrypted data.

- **Securing computation outsourcing:** This cryptographic technique enables organizations to outsource computations to third-party service providers while keeping data confidential. The service provider can perform computations on the encrypted data without having access to the plaintext.

- **Preserving machine learning (ML) privacy:** Homomorphic encryption is increasingly being explored in relation to ML, where models can be trained on encrypted data. It allows organizations to collaborate on building ML models without sharing the raw training data.

What Are the Applications of Homomorphic Encryption?

- **Securing healthcare data sharing:** In healthcare, this technique allows users to share patient data with healthcare providers or researchers securely. It enables collaborative research without compromising patient privacy.

- **Processing financial data:** Homomorphic encryption in the financial sector lets users perform computations on encrypted data. It allows financial institutions to collaborate and analyze aggregated data without exposing sensitive customer information.

- **Securing multiparty computations:** Homomorphic encryption is a key component in secure multiparty computation, where multiple parties want to compute a function over inputs while keeping them private jointly. Each party encrypts its input, and computations are performed on encrypted data.

- **Securing searches in encrypted databases:** This cryptographic technique allows you to perform secure searches on encrypted databases. Users can submit queries without revealing search details to the database owner.

Credit Card Fraud Demonstration

Real Time Fraud Detection

By creating a data set with a single transaction we can evaluate whether or not this transaction is fraudulent by encrypting it and sending it to a third party to be evaluated using an encrypted logistic regression model.

Data

This is an example of a fraudulent transaction as it appears in the trusted environment. This transaction can be encrypted just before evaluation to get real time secure predictions:

V1	V2	...	V28	Amount
-4.3980	1.3584	...	0.8496	59.0000

Encrypted Data

This is a buffered sample of the data as it appears in the untrusted environment. Data is received in an encrypted form and then prepared in the FHE context for processing:

b'RYLH\x00\x03\x05\x01\x0e\x00\x00\x00EncryptedData\x00` \xc4\td\x01\x00\x00\x00\x00\x00\x00\x00\x01\x00\x00\x00\x00\x00\x00\x00RYLH\x00\x03\x05\x01\x0f\x00\x00\x00EncryptedBatch\x00` \xc4\td\x01\x00\x00\x00\x00\x00\x00\x00\x00RYLH\x00\x03\x05\x01\x0c\x00\x00\x00CTileTensor\x00` \xc4\td\x01\x03\x00\x00\x00\...

Predicted Outcome

WARNING: Transaction is fraudulent!

Time Taken

Prediction took: 0.38909275084733963s

Westpac

CREST CISA Certified Information Systems Auditor. An ISACA® Certification bsi ISO/IEC 27001 Information Security Management CERTIFIED bsi ISO 9001 Quality Management Systems CERTIFIED Crown Commercial Service

Assentian Limited

Heart Disease Detection Demonstration

Patient Action Recommendation

In this demonstration a group of patients are evaluated against a pre-trained neural network. With the patient data being protected and highly sensitive information sending this data to a third party would be a breach of confidentiality but by encrypting it before having it evaluated we can circumvent these restrictions and avoid a breach.

Andaman7

These are the results produced by the analysis. The true patient outcome is the action the patient should hav taken in retrospect. The predicted patient outcome is the action we predict the patient should take analysing the encrypted data without knowledge of the true outcome.

True Patient Outcome	Predicted Patient Outcome
Healthy	Healthy
Healthy	Healthy
Should talk with a Dr.	Should talk with a Dr.
Should talk with a Dr.	Should talk with a Dr.
Should talk with a Dr.	Healthy
Healthy	Healthy

Searching an Encrypted Database

A database can be searched by encrypting both the dataset and the query and running a search for the encrypted query. This means that organisations can keep their data encrypted while preserving its usefulness without having to unencrypt it at the time of query thus exposing it to security risks.

Botswana

Pop. Density (per sq. mi.)

Search Database

Telstra

Results

The Pop. Density (per sq. mi.) of Botswana is 2,7

Time Spent

Time taken to encrypt the database:
0.11065087467432022s

Time taken to encrypt the query:
0.0004068836569786072s

Time taken to perform the search:
6.891290728002787s

Time taken to decrypt the result:
0.11065087467432022s

CREST CISA Certified Information Systems Auditor. An ISACA® Certification bsi ISO/IEC 27001 Information Security Management CERTIFIED bsi ISO 9001 Quality Management Systems CERTIFIED Crown Commercial Service

Assentian Limited

Andaman7 Westpac Telstra

FHE Demo

FHE Demo

- HE_Demo.ipynb
- **Title:** *Privacy-Preserving Computation with the Paillier Encryption Scheme*
- **Purpose:**
  - To show how sensitive numerical data can be **processed securely while encrypted** — enabling computation **without exposing raw values**.
- **Key ideas demonstrated:**
- **Homomorphic Encryption (HE):**
  - A special type of encryption that allows mathematical operations to be performed **directly on encrypted data**.
  - After decryption, results match what you'd get if you computed on plain data.
- **Paillier Scheme:**
  - Supports *additive* operations (you can add encrypted numbers, or multiply them by a known constant).
  - Keeps the data private throughout the process.

FHE Demo

- HE_Demo.ipynb
- **Title:** *Privacy-Preserving Computation with the Paillier Encryption Scheme*
- **Use Case Example:**
  - Enables a cloud server or data aggregator to compute **totals, averages, or statistics** on sensitive inputs **without seeing individual values** (e.g., salaries, medical metrics, financial transactions).
- **Workflow in the notebook:**
  - Generate a public/private key pair.
  - Encrypt simple numbers and vectors.
  - Perform addition and scaling *while encrypted*.
  - Decrypt to verify results match plaintext computations.
  - Demonstrate computing an average privately.
  - Handle floating-point numbers using scaling (fixed-point).
  - Measure performance and note limitations.

FHE Demo - What We Learn from the Demo

- HE_Demo.ipynb
- **Title:** *Insights and Practical Takeaways*
- **Results shown in the notebook:**
- The encrypted computations return **exact results** after decryption:
- Example:
  - E(42) + E(17) \Rightarrow 59 after decryption.
  - Secure vector sum and mean computed correctly.
  - Floating-point sums recovered accurately using scaling.
  - Performance test:
    - Encrypting and summing 200 values took ~20 seconds (CPU-based demo).
    - Highlights the **computational cost** of encryption.

FHE Demo - What We Learn from the Demo

- HE_Demo.ipynb
- **Title:** *Insights and Practical Takeaways*
- **Key limitations and considerations:**
- Paillier supports **only addition and scaling**, not full multiplication between ciphertexts.
- Each value is encrypted separately → **storage and speed trade-offs** for large datasets.
- Handling of decimals requires **manual scaling**, which can introduce rounding limits.

FHE Demo - Homomorphic_Encryption.ipynb

- **Title:** *Computing on Encrypted Data: A Tiny FHE Example*
- **Purpose:** To show, in a simplified and educational way, how a **fully homomorphic encryption system** works — where encrypted data can still be used for computation **without ever being decrypted**.
- **What happens in the notebook:**
- **Setup:**
  - Defines tiny parameters for a toy version of an encryption scheme based on *learning with errors (LWE)* — far too small for real security, but great for illustrating the idea.
- **Key generation:**
  - A **secret key** (a random odd number) is created.
  - A **public key** is derived from it — a list of numbers that hide the secret key behind small random "noise."
- **Encryption:**
  - To encrypt a bit (0 or 1), the code randomly combines parts of the public key and adds noise so the message is hidden within a large integer.
- **Decryption:**
  - Uses the secret key to remove the noise and extract the original message bit (0 or 1).
- **Testing:**
  - Runs 1,000 trials to confirm messages encrypt and decrypt correctly.

Technical Challenges and Advancements

Computational Complexity: Homomorphic computations can be computationally expensive, leading to slower processing times compared to traditional methods.
Advancements: Research focuses on optimizing HE schemes and utilizing specialized hardware accelerators to improve performance.

Limited Functionality: Current FHE schemes might not support all desired operations or data types.
Advancements: Ongoing research explores expanding the capabilities of FHE schemes to handle more complex computations and data structures.

Multi-Party Computation: Collaborative Analysis Without Sharing Data

Multi-Party Computation: Collaborative Analysis Without Sharing Data

What is multi-party computation (MPC)?

Multi-party computation (MPC) is a subfield of cryptography focused on secure computation that allows multiple participants to perform calculations while keeping individual data secret. Each participant will only see the final result, not the inputs of others. MPC combines cryptographic protocols for computation, verification (to detect cheating), and privacy preservation.

Imagine three companies tasked with calculating their average revenue without disclosing the actual numbers to each other. MPC allows them to compute the average, with each company seeing only the final result, not the other companies' revenue.

While both zero-knowledge proofs (ZKPs) and multi-party computation (MPC) belong to privacy-preserving cryptography, they serve distinct purposes. ZKPs allow a party to prove knowledge of a secret without revealing it (e.g., proving identity without disclosing any sensitive credentials), even in multi-party settings. MPC, however, enables multiple parties to jointly compute a function over their private inputs without exposing the inputs themselves.

The key breakthrough of multi-party computation is that parties can collaborate on calculations without revealing their data to anyone.

Core concepts

Privacy preservation

Multi-party computation ensures input privacy through mathematical guarantees. The protocol mathematically prevents participants from learning others' inputs while still allowing computation on those inputs.

Distributed trust

No single party holds all the data or controls the computation. The security of multi-party computation comes from distributing trust across multiple participants, making it resistant to individual compromises.

What is MPC (multi-party computation) and how does it work?

In an MPC, a given number of participants each possess a piece of private data (d1, d2, ..., dN). Together, the participants can compute the value of a public function on that private data: F(d1, d2, ..., dN) while keeping their own piece of data secret.

For example, let's imagine three people, John, Rob, and Sam, want to find out who has the highest salary without revealing to each other how much each of them makes – this is actually a classic example of multi-party computation, known as **The Millionaire's Problem**. Using simply their own salaries (d1, d2, and d3), they want to find out which salary is the highest and not share any actual numbers with each other. Mathematically, this translates to them computing:

F(d1,d2,d3) = max(d1,d2,d3)

If there were some trusted third party (i.e. a mutual friend who they knew could keep a secret), they could each tell their salary to that friend and find out which of them makes the most, AKA F(d1,d2,d3), without ever learning the private info. The goal of MPC is to design a protocol, where, by exchanging messages only with each other, John, Rob, and Sam can still learn F(d1,d2,d3) without revealing who makes what and without having to rely on an external third party. They should learn no more by engaging in the MPC than they would have by interacting with their trustworthy mutual friend.

The importance of MPC

MPC solves a specific problem in data privacy: how to compute sensitive data without exposing it. This matters for several reasons:

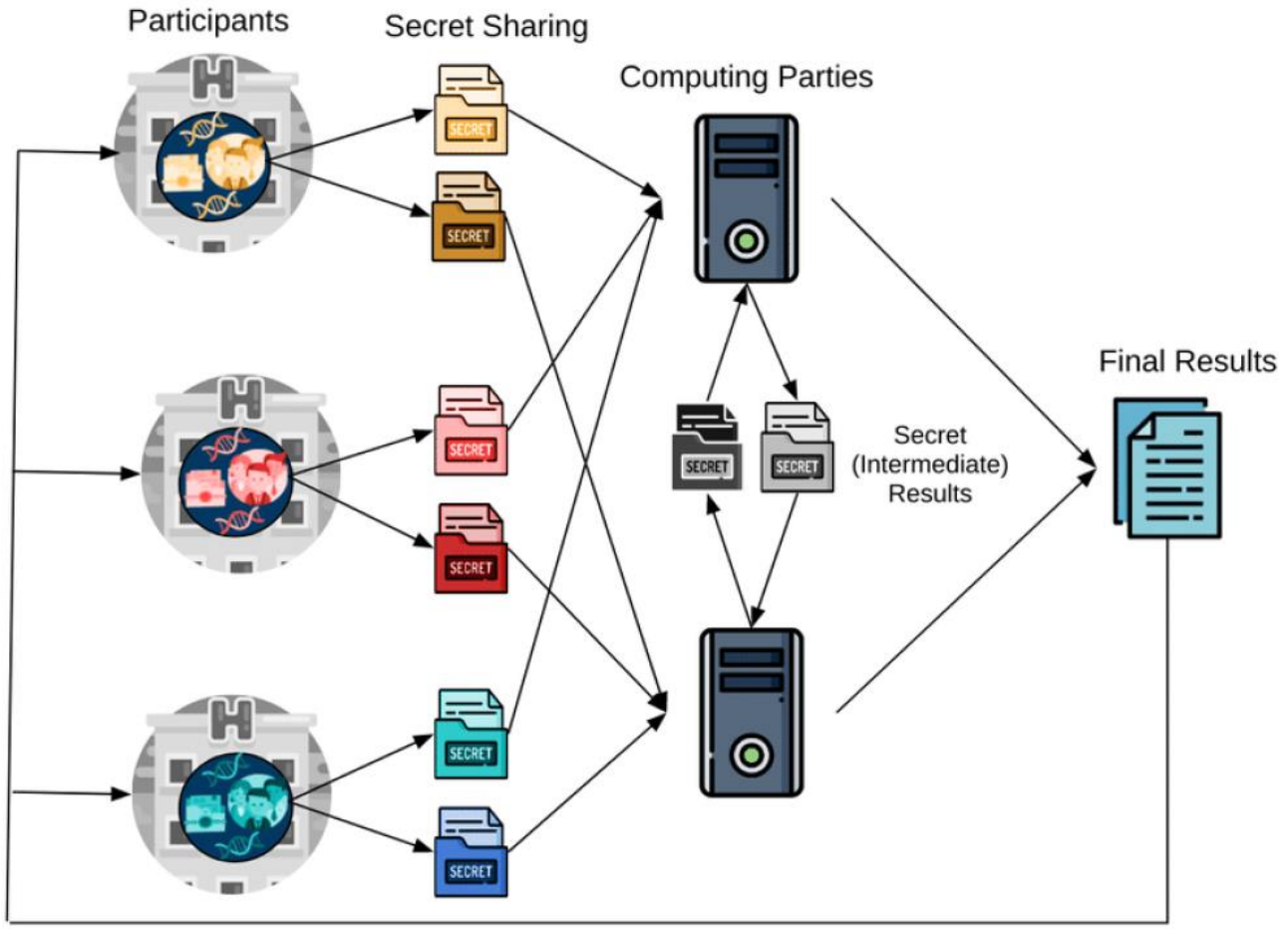
1. **Data privacy laws**: Organizations must analyze data while complying with regulations such as the General Data Protection Regulation (**GDPR**) and the California Consumer Privacy Act (**CCPA**).

2. **Cross-organization collaboration**: Companies can work together without sharing confidential information.

3. **Single point of failure**: Traditional systems that collect and centralize data in a single location create security risks.

4. **Trust minimization:** MPC removes reliance on a single trusted third party. For example, organizations can collaborate without sharing raw data, and cryptographic systems (like ZKP trusted setups) can generate public parameters securely without a central authority.

What is MPC (multi-party computation) and how does it work?



What is an adversary in MPC?

In MPC protocols, an "**adversary**" refers to a participant (or group of participants) who attempts to break the protocol's security. Adversaries are bad actors in the system. They could be:

- External attackers trying to learn private information
- Participants in the protocol who don't follow the rules
- Compromised parties (like hacked servers)

Therefore, MPC protocols are designed around two foundational questions:

1. **Adversarial settings**: How many participants can be corrupt before a protocol's security guarantee fails? *(**Note**: Often referred to simply as "**settings**" in academic papers).* **Example**: A protocol designed for a dishonest majority setting (*explained below*) is mathematically guaranteed to remain secure even if 4 out of 5 parties are corrupted.

2. **Adversarial behavior**: What tactics can parties use to attack the system? **Example**: Passive observers might be able to infer private data, while active attackers could falsify inputs or collude.

These questions define the protocol's resilience to collusion (e.g., multiple corrupted parties secretly sharing data to breach privacy) and its ability to handle adversarial actions like eavesdropping (intercepting messages to infer private inputs) or sabotage (deliberately disrupting computations).

Semi-honest security model (passive / honest-but-curious)

What it means:

The semi-honest security model assumes adversaries follow the protocol rules but attempt to infer private information from exchanged messages. A protocol that is secure against semi-honest adversaries (known as a *semi-honest secure* protocol) ensures that parties **can't** infer private inputs, even if they passively analyze protocol interactions.

How it works:

Semi-honest secure protocols use cryptographic primitives like garbled circuits and secret-sharing schemes (*discussed later*) to ensure no single party can reconstruct private data without collaboration (e.g., obtaining enough secret shares).

Use cases:

1. **Banks using MPC for fraud detection**: Semi-honest security allows multiple banks to collaboratively analyze transaction data and identify fraud patterns without exposing customer records. Semi-honest protocols provide security against passive adversaries while maintaining the computational efficiency required for processing large-scale datasets.

2. **Medical research collaborations**: Semi-honest protocols allow hospitals to jointly analyze patient genomes while preserving privacy. By minimizing computational overhead, these protocols remain practical for data-heavy tasks like genomic studies.

Benefits of semi-honest security:

- **Efficiency**: Generally requires fewer cryptographic verification steps than malicious-secure protocols (e.g., no zero-knowledge proofs or cut-and-choose).

- **Lower overhead**: Reduced computational costs compared to active-adversary models.

- **Scalability**: Ideal for applications where participants are incentivized to cooperate (e.g., mutually beneficial analytics)

Malicious (active) security model

What it means:

The malicious security model assumes adversaries may arbitrarily deviate from the protocol: sending fake inputs, tampering with computations, or aborting prematurely to disrupt results. A *malicious-secure* protocol guarantees privacy and correctness even if some parties cheat, ensuring outputs are valid and inputs remain confidential.

How it works:

The malicious-secure protocol includes additional cryptographic verification steps (e.g., zero-knowledge proofs, cut-and-choose, authenticated secret sharing MACs) that mathematically detect and prevent cheating. It forces parties to prove they follow the protocol correctly at each step, guaranteeing input privacy and computation integrity, even against active attackers.

Use cases:

1. **Voting systems**: Malicious security ensures no party can submit invalid ballots (e.g., votes for non-existent candidates) or manipulate tallying. For example, ZK proofs prove that votes are valid (e.g., for registered candidates) without revealing individual choices.

2. **Cryptocurrency**: Threshold signatures require malicious security to prevent adversarial parties from biasing key generation or stealing funds. Checks ensure that private keys are never reconstructed in a single location.

Benefits of malicious-secure security:

- **Stronger security**: Privacy and correctness are guaranteed even if adversaries sabotage the protocol.

- **Deterrence**: Cheating is either detected (e.g., honest parties abort the protocol if invalid ZK proofs are detected) or rendered cryptographically impossible.

- **High-stakes suitability**: This is essential for applications like financial settlements, where errors or leaks have severe consequences.

Secret sharing in multi-party computation

Secret sharing allows splitting a secret into multiple pieces, called shares, where:

- Each participant receives a share
- Individual shares reveal nothing about the secret
- Combining enough shares reconstructs the secret

In a threshold secret-sharing scheme:

- **n**: The total number of shares (pieces) a secret is split into (e.g., 5).
- **t**: The threshold - minimum number of shares needed to reconstruct the secret (e.g., 3)

MPC Video

Secure Multi-Party Computation: Protecting Data Privacy in AI

Secure Multi-Party Computation (MPC) is a powerful tool for safeguarding data privacy in AI systems. Here's what you need to know:

- MPC allows multiple parties to compute together while keeping individual data private
- It addresses key AI privacy concerns like data breaches, regulatory compliance, and trust issues
- MPC offers advantages over other privacy methods for AI applications

Feature	MPC	Homomorphic Encryption	Differential Privacy	Zero-Knowledge Proofs
Data Protection	Very good	Very good	Good	Very good
Speed	Fast	Slow	Fast	Slow
Works with Big Data	Yes	No	Yes	No

Secure Multi-Party Computation: Protecting Data Privacy in AI

Key benefits of MPC for AI:

- Enables private AI training across organizations
- Allows secure data sharing and collaboration
- Protects sensitive data during AI computations
- Helps meet data protection regulations

While MPC faces some challenges like complexity and performance, ongoing research is improving its capabilities for AI applications. As privacy concerns grow, MPC will likely play an increasingly important role in responsible AI development.

Secure Multi-Party Computation: Protecting Data Privacy in AI

MPC helps keep data private when training AI. It lets different groups work together on AI without showing their private information. This is good for sensitive data like health records or bank details.

MPC Benefit	Description
Private training	Groups can train AI together without sharing raw data
Safe collaboration	Sensitive info stays hidden during AI development

Secure Multi-Party Computation: Protecting Data Privacy in AI

MPC allows AI to learn from many sources while keeping data private. This makes AI better without risking privacy.

Feature	Outcome
Joint computation	AI learns from various data sources
Privacy protection	Each group's data remains secret

Secure Multi-Party Computation: Protecting Data Privacy in AI

MPC keeps AI predictions and results private. This is key for areas like healthcare and finance where privacy is a must.

Aspect	Benefit
Confidential outputs	Predictions stay secret
Limited access	Only authorized people see results

Secure Multi-Party Computation: Protecting Data Privacy in AI

With MPC, groups can make sure data is correct without actually looking at it. This builds trust and keeps information safe.

MPC Capability	Advantage
Data verification	Ensure accuracy without exposure
Privacy preservation	Check data quality while maintaining secrecy

Adding MPC to AI systems

What you need to use MPC

To add MPC to AI systems, you'll need these key parts:

Component	Purpose
Safe communication channel	Lets groups work together on AI training
Math tools for privacy	Keeps data safe during sending and use
Ways to hide and show data	Makes sure data stays private while being used

Adding MPC to AI systems

What you need to use MPC

When using MPC in AI, you might face these issues:

Problem	Solution
Data leaks	Hide data and control who can see it
Hard-to-use MPC tools	Use ready-made MPC tools that are easier

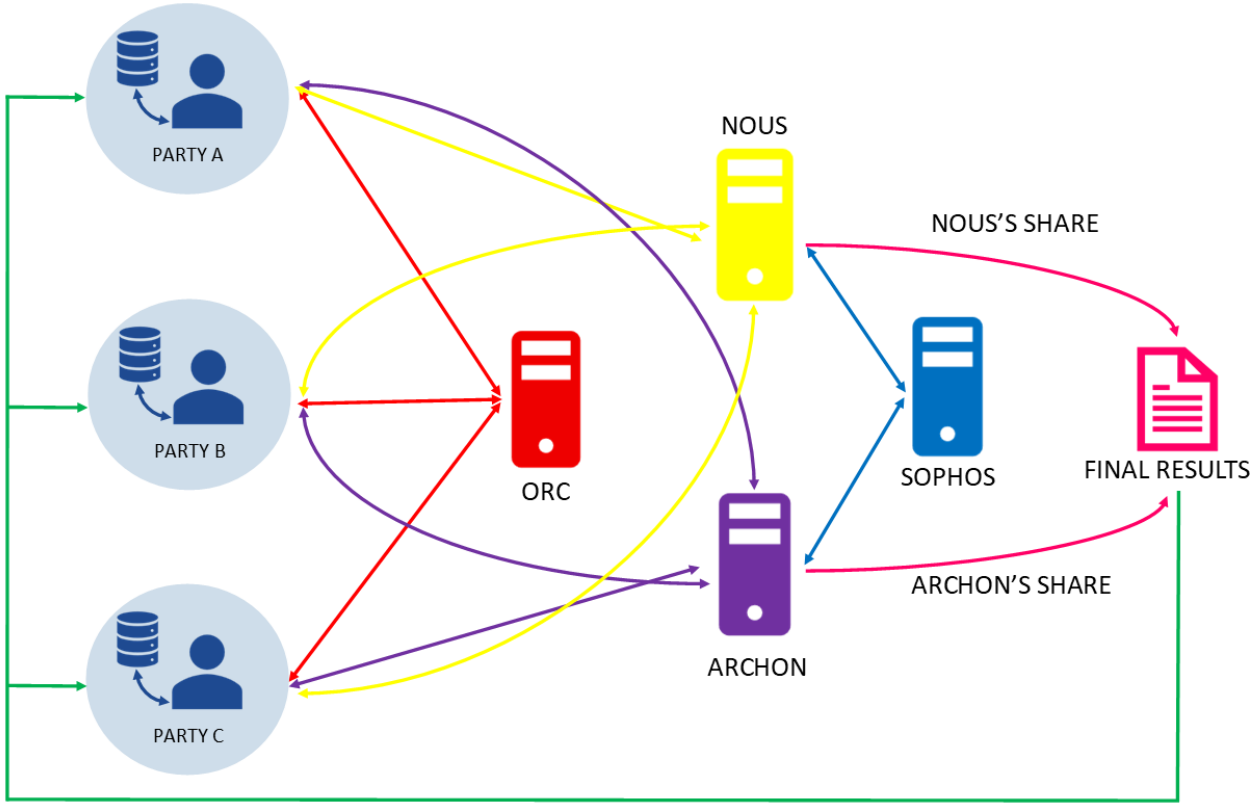
Adding MPC to AI systems

What you need to use MPC

https://github.com/rdragos/awesome-mpc

- Theory
  - Books
  - Courses
  - Tutorials
- Misc
- Software
  - Frameworks
  - Primitives
  - Protocols
  - Tools
  - Retired software
- Workshops

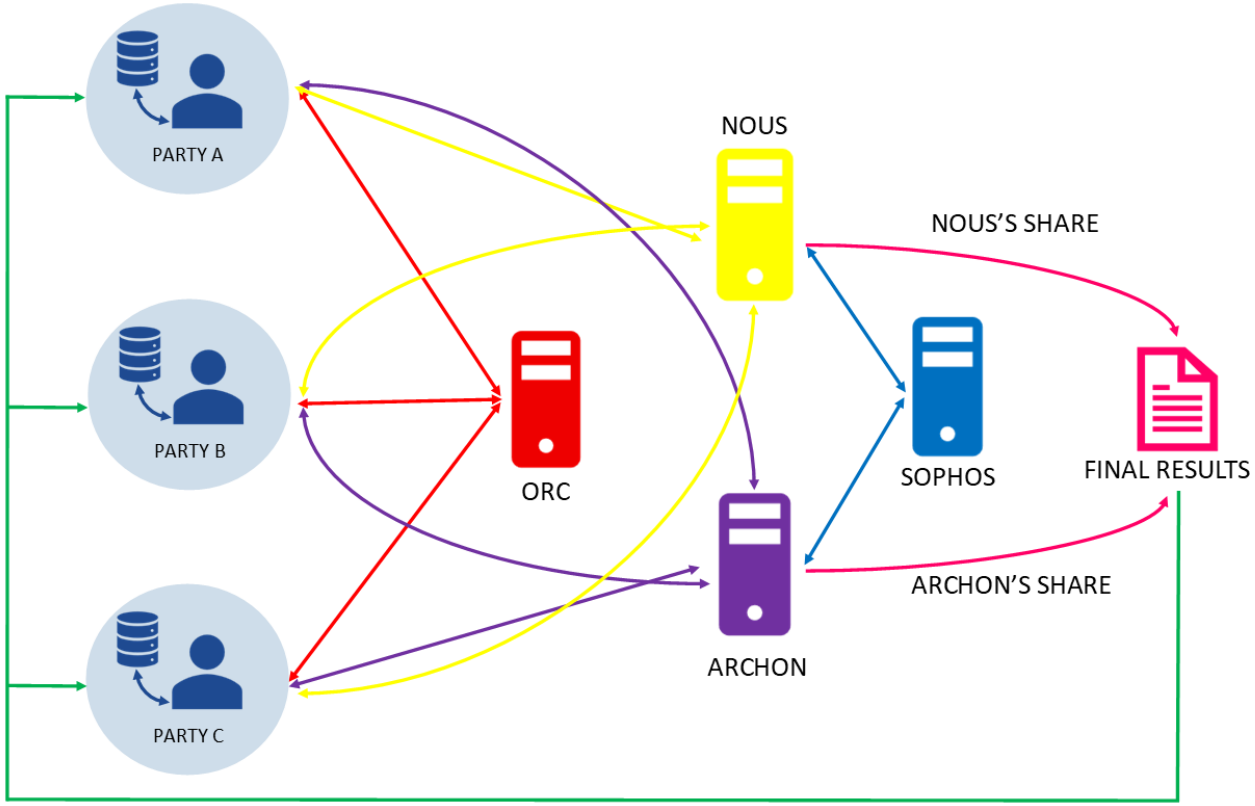
Multi-Party Computation: Collaborative Analysis Without Sharing Data



Input sharing (secret splitting)

- A secure coordination node (ORC).
- Once all 3 parties are ready ORC will send a signal to the parties to begin the computation.
- Each party then splits their private data into masked secret shares distributed between NOUS and ARCHON.
- Individual shares are meaningless on their own, only a sufficient subset can reconstruct the original value.

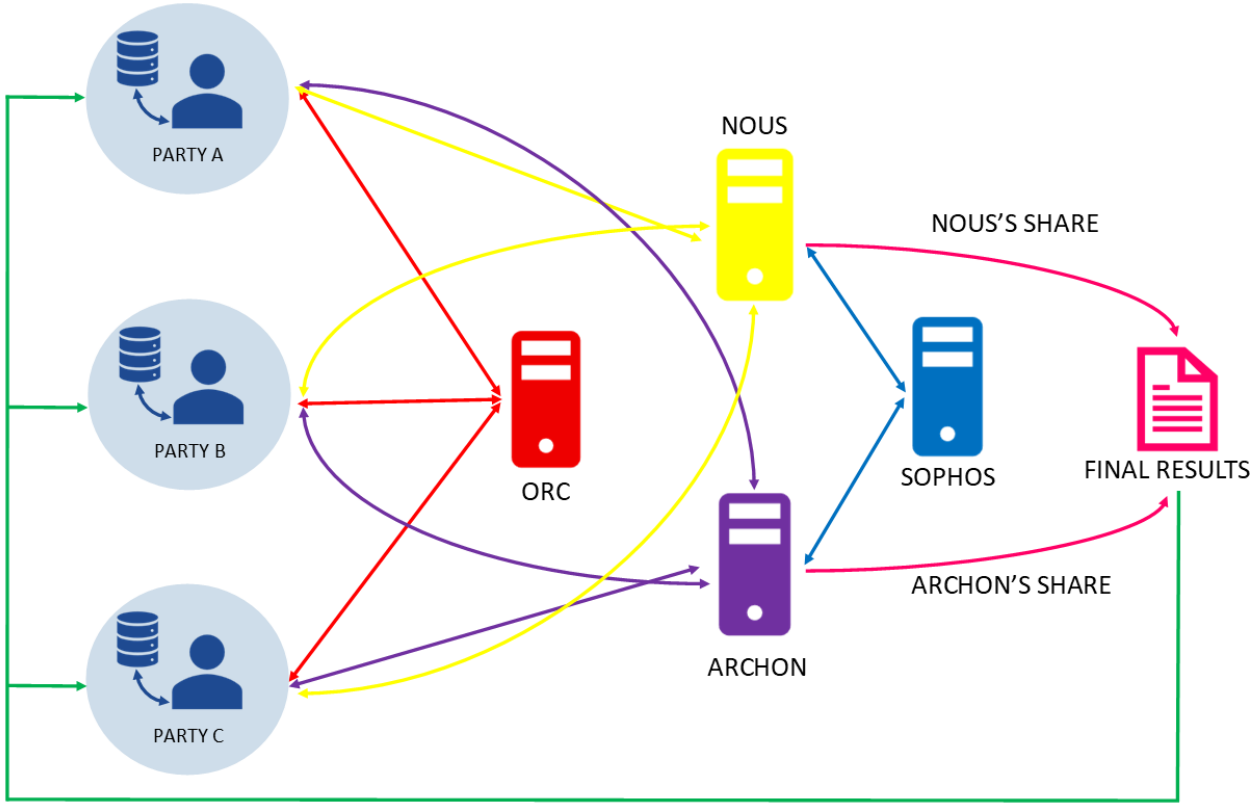
Multi-Party Computation: Collaborative Analysis Without Sharing Data



Secure computation on shares

- The nodes perform computations on their shares without reconstructing the original values.
- If SOPHOS is required for comparision computations. Each node will send its masked shares to SOPHOS to be combined but never reconstructing the original values.
- The comparison is performed then the relative shares are sent back to NOUS and ARCHON.

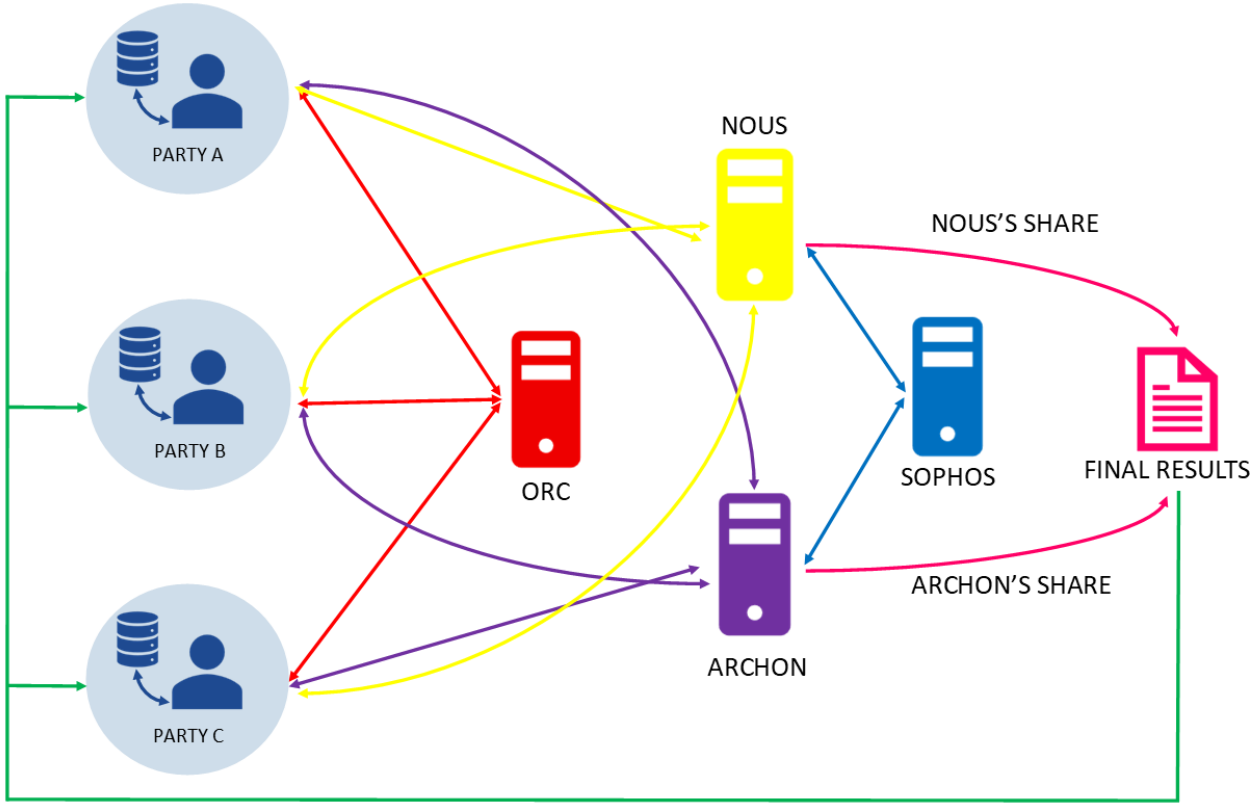
Multi-Party Computation: Collaborative Analysis Without Sharing Data



Result reconstruction & release

- The masked shares results from NOUS and ARCHON are returned to all the parites where they are reconstructed and demasked to reveal the true result.
- Only the agreed-upon outcome is revealed, no extra inputs are leaked.

Multi-Party Computation: Collaborative Analysis Without Sharing Data



DEMO

Multi-Party Computation: Collaborative Analysis Without Sharing Data

- Multi-party computation (MPC) in commodity trading allows multiple parties, like buyers and sellers, to jointly compute a market-clearing price or other outcomes without revealing their sensitive, private data to each other.
- This cryptography-based approach enables collaboration while protecting individual costs, strategies, and bids, fostering fair pricing and trust in competitive markets

cme
Chicago Mercantile Exchange

Multi-Party Computation: Collaborative Analysis Without Sharing Data

Key applications and benefits

- **Market-clearing price calculation:** Instead of revealing their own costs or pricing strategies, companies can use MPC to securely compute a market-clearing price that balances supply and demand across all participants.
- **Secure auctions:** Participants can submit bids without revealing them to competitors, ensuring a fair and competitive auction process where only the final result is public.
- **Data-driven collaboration:** Banks or other financial institutions can collaborate on risk assessments or fraud detection without sharing sensitive customer or transaction data with each other

cme
Chicago Mercantile Exchange

CREST
CISA Certified Information Systems Auditor. An ISACA® Certification
bsi ISO/IEC 27001 Information Security Management CERTIFIED
bsi ISO 9001 Quality Management Systems CERTIFIED
Crown Commercial Service

Assentian Limited

Multi-Party Computation: Collaborative Analysis Without Sharing Data

Example scenario:

- Three friends (Alice, Bob, Charlie) want to compute their **total savings** — but **no one** wants to reveal their individual balances.
- **Core Idea — Additive Secret Sharing:**
  - Each private value is split into random "shares."
  - Each participant gets **one share** of everyone's data.
  - No single share reveals anything.
  - When shares are **combined**, the true value is reconstructed.
- **TinySMPC = Educational SMPC in Python**
  - Minimal, easy to read.
  - Implements key SMPC primitives:
  - PrivateScalar → private data
  - SharedScalar → secret-shared data
  - VirtualMachine → participant's local node

Multi-Party Computation: Collaborative Analysis Without Sharing Data

Example workflow:

- Create 3 VirtualMachines (Alice, Bob, Charlie).
- Each has a private balance (PrivateScalar).
- Use .share() to split and distribute secret shares.
- Compute **sum**, **multiplication**, or **other operations** directly on encrypted shares.
- Use .reconstruct() to recover the **public result**.

Limitations & Insights:

- Assumes *honest-but-curious* participants.
- Requires all parties online.
- Communication-intensive → slower over networks.
- Great for understanding core SMPC concepts.

Differential Privacy

Harvard John A. Paulson
School of Engineering
and Applied Sciences

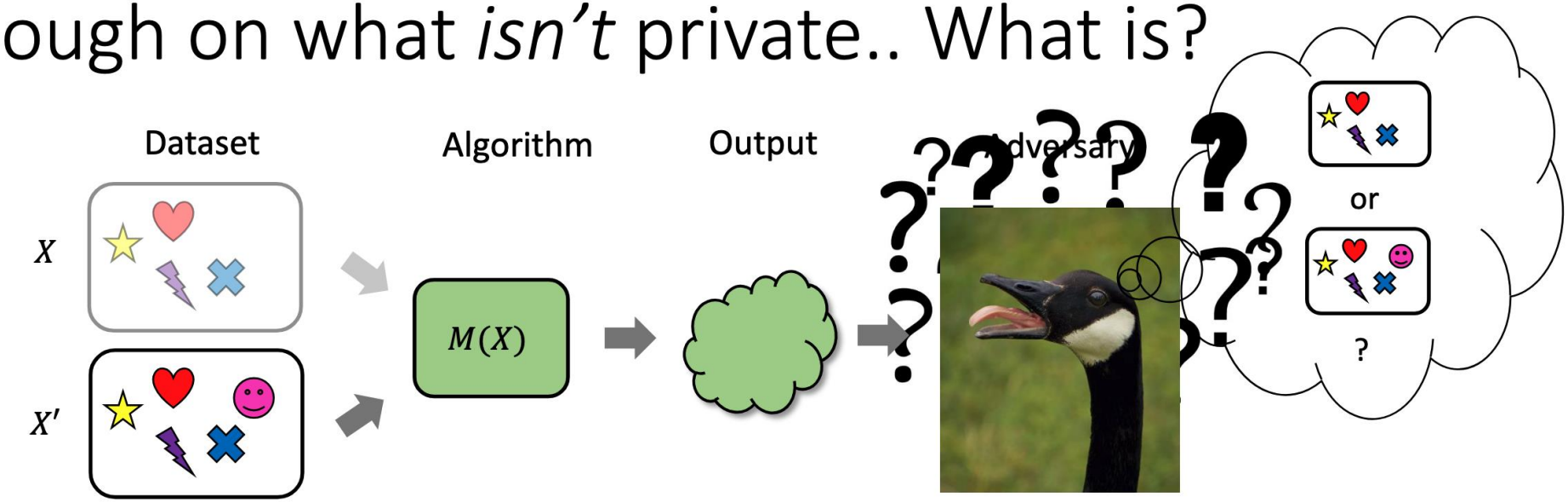
What is Differential Privacy?

Differential privacy is a rigorous mathematical definition of privacy. In the simplest setting, consider an algorithm that analyzes a dataset and computes statistics about it (such as the data's mean, variance, median, mode, etc.). Such an algorithm is said to be differentially private if by looking at the output, one cannot tell whether any individual's data was included in the original dataset or not. In other words, the guarantee of a differentially private algorithm is that its behavior hardly changes when a single individual joins or leaves the dataset -- anything the algorithm might output on a database containing some individual's information is almost as likely to have come from a database without that individual's information. Most notably, this guarantee holds for *any* individual and *any* dataset. Therefore, regardless of how eccentric any single individual's details are, and regardless of the details of anyone else in the database, the guarantee of differential privacy still holds. This gives a formal guarantee that individual-level information about participants in the database is not leaked.

The definition of differential privacy emerged from a long line of work applying algorithmic ideas to the study of privacy (Dinur and Nissim `03; Dwork and Nissim `04; Blum, Dwork, McSherry, and Nissim `05), culminating with work of Dwork, McSherry, Nissim, and Smith `06.

See our educational materials for more detail about the formal definition of differential privacy and its semantic guarantees.

Enough on what *isn't* private.. What is?



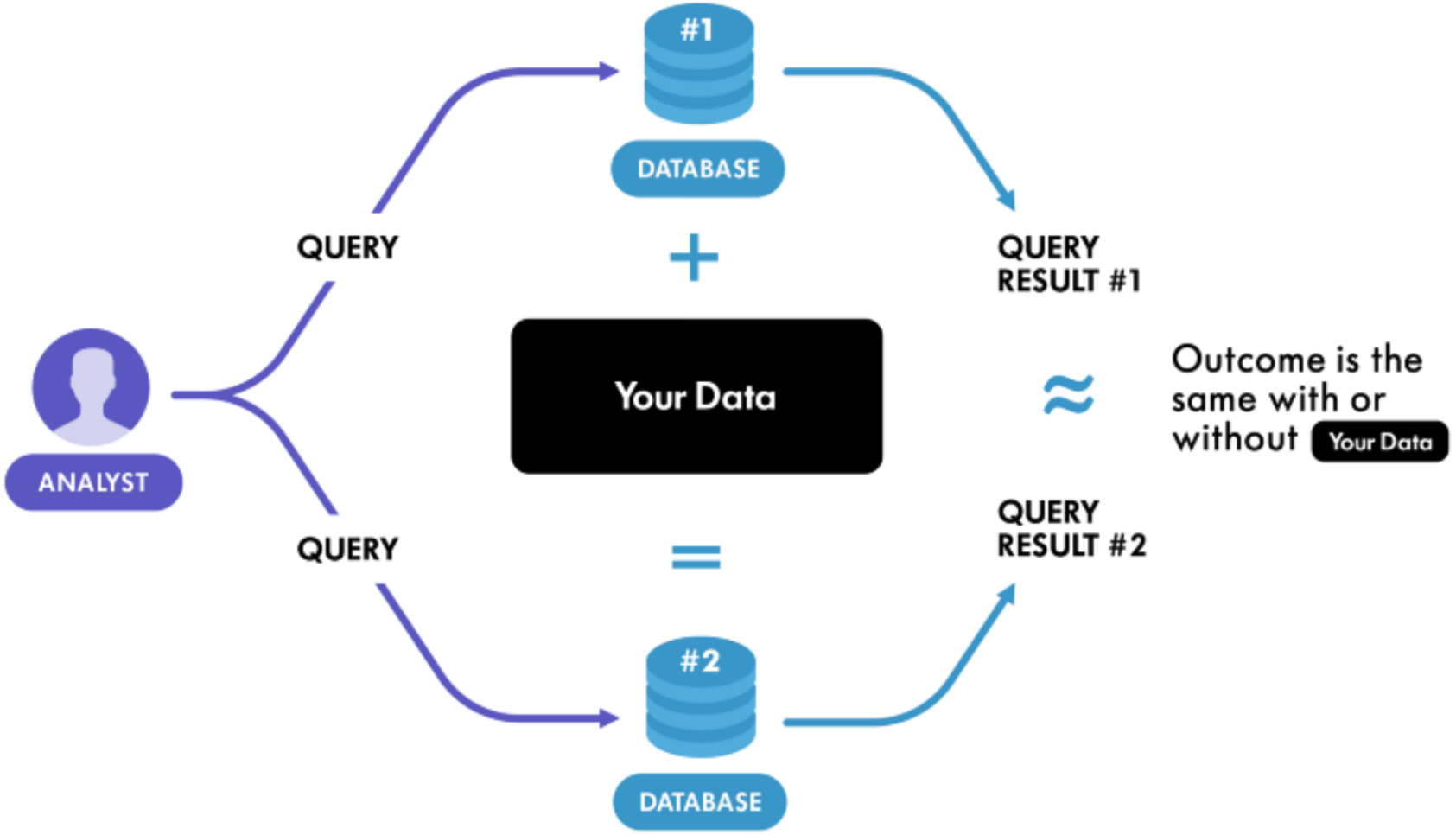
"An algorithm is differentially private if its distribution over outputs doesn't change much after adding/removing one point."

Differential Privacy (Informal)

"An algorithm is differentially private if its distribution over outputs doesn't change much after adding/removing one point."

Why is this a reasonable notion of privacy?

- Dropping a user's datapoint is unlikely to change the output
- Thus looking at the output, can't tell if a user was in the dataset or not
- If you can't even know if a user is present, you can't know their data
- E.g., protects against database reconstruction attacks (and much more!)



What are the different types of differential privacy?

There are two main types of differential privacy: global and local. Global differential privacy (GDP) applies noise to the output of an algorithm that operates on a dataset, such as a query or a model. Local differential privacy (LDP) applies noise to each individual data point before sending it to an algorithm, such as a survey or a telemetry system.

Global Differential Privacy (GDP) Explained

- Centralized Noise Addition: In Global Differential Privacy, noise is added by a trusted data curator after collecting the raw data, before releasing any analysis or query results.
- Trust Model: GDP assumes the data curator is fully trusted to handle raw data securely and to add the appropriate noise to protect individual privacy before sharing outputs.
- Goal: It aims to provide strong privacy guarantees by ensuring that the inclusion or exclusion of any single individual's data in the dataset minimally impacts the results, making it hard to infer private information.
- Accuracy Focus: Because noise is added only once at the global level, this approach often yields more accurate results compared to local privacy methods that add noise at the data source.
- Use Cases: Common in scenarios like government statistics, healthcare research, and centralized machine learning, where a trusted entity manages sensitive data and releases differentially private aggregate insights.

Local Differential Privacy (LDP) Explained

- Privacy at the Source: In LDP, noise is added directly to each individual's data before it is collected or sent to any server, meaning raw sensitive data is never shared.
- No Trusted Aggregator Needed: Unlike global differential privacy, LDP does not rely on a trusted central party. Each user protects their own privacy independently.
- Strong Privacy Guarantees: Even if the data collector is compromised or malicious, the noise added locally ensures privacy is maintained for each individual's data.
- Trade-off: Increased Noise: Because noise is added at the individual level, the aggregate data is typically noisier, potentially lowering overall accuracy compared to global differential privacy.
- Real-World Deployments: LDP has been successfully deployed in practice by large companies like Google (RAPPOR) and Apple for collecting usage statistics while preserving user privacy.
- Use Cases: Commonly used in federated learning, telemetry data collection, and scenarios where user trust in the data collector is limited or nonexistent.

What does DP protect against?

- Database reconstruction
  - Finding a user's private data
- Membership inference
  - Determining whether or not a user was in the dataset
- Learning anything about a user that can't be inferred w/o them

What *doesn't* DP do?

- Important: does **not** prevent inferences (statistics/machine learning)
  - (Public) smoker participates in (differentially private) study investigating whether smoking causes cancer
  - Reveals that smoking causes cancer! Smoker's insurance premiums increase!
  - Was their (differential) privacy violated?
  - No: smoking → cancer could be inferred whether or not they participated
  - Differential privacy: outcome of algorithm is similar, whether or not someone participates
- Not appropriate when individual identities are important
  - "Private" contact tracing

Differential Privacy

On to the algorithms!

The Laplace Mechanism

The Laplace mechanism for differential privacy can be explained in a simple way as follows:
Imagine you want to share information about a group without revealing details about any one individual.

- The Laplace mechanism does this by adding a kind of “fuzzy noise” to the answers, so the results are slightly blurred.
- This noise is carefully chosen to make it hard to tell whether any single person’s data was included or not, protecting individual privacy while still providing useful overall information.
- The amount of noise added depends on how much the answer could change if one person’s data were added or removed—balancing privacy with accuracy.

The Laplace Mechanism

Example: Counting

Suppose you want to share the number of people in a room, but keep individual presence private.

- Instead of saying the exact count (e.g., 20 people), you add some “random noise” from a special Laplace distribution, say you report 20 ± 2.
- This noise hides whether any specific person was counted, so no one’s presence or absence can be confidently inferred.
- The amount of noise depends on how much the count could change by adding or removing one person, balancing privacy and accuracy.

This example visually conveys the core idea of the Laplace mechanism clearly for any audience while highlighting its privacy utility trade-off

Exponential Mechanism

The exponential mechanism in differential privacy can be explained simply as follows:

- When you want to pick the best answer from many options but still keep individual data private, the exponential mechanism helps by making higher-quality answers more likely to be chosen, without always picking the single best one.
- It adds a “controlled randomness” that favors better options but still keeps some uncertainty, so no single individual’s data can be pinpointed from the selection.
- This way, it balances privacy and usefulness by choosing outputs based on how good they are, while protecting data privacy by not revealing too much about any individual in the dataset.
- In essence, the exponential mechanism privately picks good answers by making them more likely but never certain, ensuring privacy and utility together

Exponential Mechanism

Example: Choosing a Winner Privately

- Imagine running a private election with three candidates and sensitive votes from people.
- Instead of picking the candidate with the most votes outright, the exponential mechanism picks a winner with probabilities that favor those with higher vote counts, but not always the highest.
- For example, if candidates have **roughly 20, 19, and 15 votes**, the mechanism might pick the winner with probabilities like **40%, 36%, and 24%, respectively.**
- This randomness protects voter privacy because the result doesn't reveal exact vote counts but still tends to select popular candidates.

This example clearly shows how the exponential mechanism balances privacy and accuracy by controlled random selection, making it easy to understand on a slide

Properties of Differential Privacy

Post-Processing

- Once data is processed with a differentially private mechanism, any further processing or analysis done on its output cannot weaken the original privacy guarantees — the privacy protection is preserved regardless of what you do with the output.
- This means you can safely transform, combine, or analyze the differentially private results without risking exposure of individual information.
- Post-processing also ensures robustness against attacks using additional auxiliary information; no matter what extra knowledge an attacker has, they cannot extract more individual data beyond the original differential privacy guarantee.
- This property enables multiple uses and refinements of private outputs (like noise reduction or aggregation) without compromising privacy.

Group Privacy

- Differential privacy guarantees privacy protection for single individuals, but this guarantee extends gracefully to groups of individuals: if individuals change, the privacy loss scales roughly by a factor of *k*.
- Formally, if a mechanism is differentially private for one person, then it is approximately differentially private for a group of people, meaning stronger noise or privacy budget reduction is needed to protect larger groups.
- This property helps quantify how privacy degrades when considering multiple individuals together and guides setting the privacy parameters based on group size.
- It also underpins the composition property, where multiple queries or mechanisms combining influence on the same dataset accumulate overall privacy cost.

Group Privacy

- *In differential privacy k means:*
  - The number of individuals or records considered together in a group when discussing group privacy properties.
  - It represents the size of the group whose combined data changes might affect the privacy guarantee.
  - More broadly, *k* is used to quantify how privacy loss scales when multiple individuals' data are changed or considered simultaneously.
    - For example, if a mechanism ensures differential privacy for one individual, it ensures appro$k\varepsilon$ nately differential privacy for a group of size *k*.
  - This means the privacy guarantee degrades linearly with the group size , requiring more noise or a smaller privacy budget to maintain the same level of privacy for larger groups.

The privacy budget

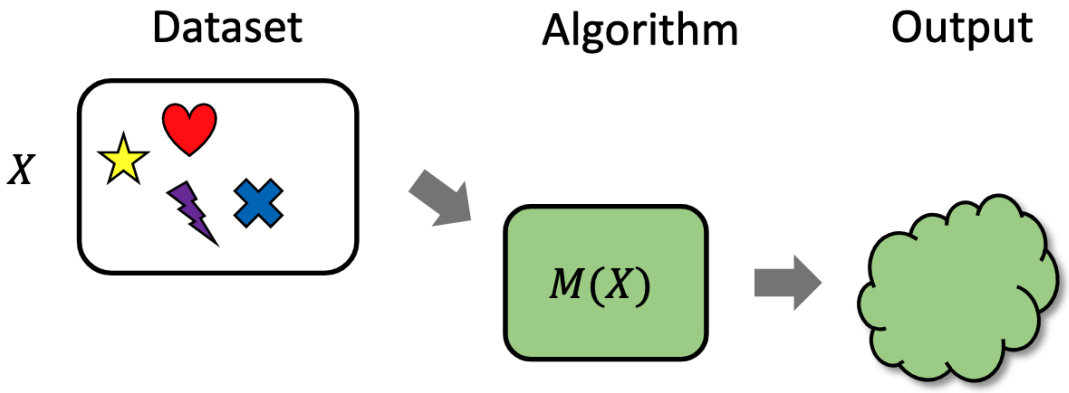
It sounds like a fancy name, but it's quite easy to understand: privacy losses are cumulative. This cumulative property is a direct consequence of the **composition theorem**. In a nutshell, with every new query that is made to a database, additional information about the sensitive data is released, hence the pessimistic view of the composition theorem, where the worst-case scenario is assumed: the same amounts of leakage happens with every new response. For strong privacy, we want the privacy loss to be minimum.

While leveraging Differential Privacy, and knowing that privacy losses are cumulative, data curators should enforce a maximum privacy loss — **the privacy budget**.

The pros

- DP provides a mathematically provable measure of user privacy. It guarantees that the result of a survey is independent of the presence of any particular individual;
- DP can be used as a defense mechanism against many types of attacks for example reconstruction attacks, linkage attacks, etc.;
- DP ensures stronger privacy for bigger datasets — the bigger the better.

Challenge



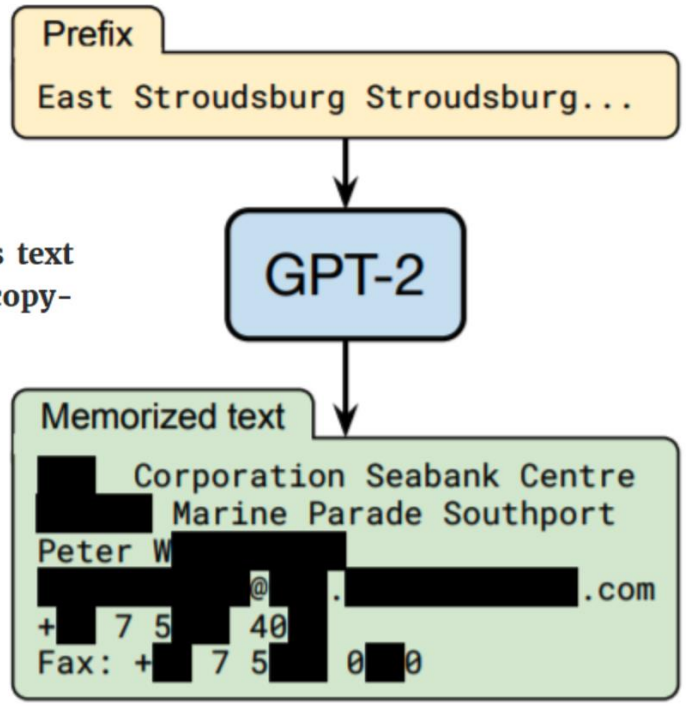
Output may still allow for inference on what is in the original dataset

- Overfitting: If the model memorizes specific data points or patterns from the training set instead of learning generalizable patterns, the outputs may inadvertently encode unique or sensitive information from the training data, allowing inference about original data points.
- Model Inversion Attacks: Attackers can use the model's predictions or outputs to reconstruct or approximate sensitive features of the training dataset. For example, by querying the model with crafted inputs and analyzing outputs, one can infer private information about the training data.
- Coefficients and Feature Contributions: In interpretable models such as linear regression, the model coefficients directly reflect relationships between dataset features and outputs, allowing insights into the training dataset's underlying distributions and feature importance.

Machine Learning Models are Vulnerable!

- Trained on very large datasets
- Can be coerced to reproduce training data verbatim!

We focus on GPT-2 and find that at least 0.1% of its text generations (a very conservative estimate) contain long verbatim strings that are "copy-pasted" from a document in its training set.



- Personal information, copyrighted content

Below, we prompt GPT-3 with the beginning of chapter 3 of *Harry Potter and the Philosopher's Stone*. **The model correctly reproduces about one full page of the book** (about 240 words) before making its first mistake.

Private Machine Learning

Private Machine Learning

- Sensitive training data
- Train a machine learning model without leaking too much info
- Can we use the ideas we know about differential privacy?

Private Machine Learning

How Differential Privacy Enhances Privacy in Machine Learning

- Protection Against Data Leakage: Differential privacy (DP) introduces controlled randomness into the training process, ensuring that the model's outputs do not reveal specific information about individual data points, thus reducing risks of data leaks or memorization.
- Resilience to Membership and Attribute Inference Attacks: By limiting how much information a model leaks about any single training example, DP makes it harder for attackers to determine if a particular individual's data was used in training or to infer sensitive attributes, thus defending against membership and attribute inference attacks.
- Guarantees Privacy Even with Auxiliary Information: Unlike traditional anonymization techniques, DP guarantees that an attacker with auxiliary knowledge cannot learn more about any individual, even when combining the model's outputs with other sources.
- Facilitates Privacy-Utility Trade-offs: With adjustable parameters (like epsilon), privacy levels can be tuned according to needs, balancing model accuracy with privacy protection, thereby making models safer without sacrificing too much utility.
- Supports Federated and Distributed Learning: DP is compatible with federated learning, where models are trained locally on devices, further reducing privacy risks by never accessing raw data centrally, and providing formal privacy guarantees throughout.

What are the applications of differential privacy?

Differential privacy has many applications in various domains, such as healthcare, social science, education, and business. For example, differential privacy can be used to protect the privacy of patients' medical records while enabling researchers to analyze them for insights.

Differential privacy can also be used to protect the privacy of students' test scores while allowing educators to evaluate their performance. Differential privacy can also be used to protect the privacy of customers' preferences while allowing businesses to personalize their services.

What is differential privacy image classification?

Differential privacy image classification is the task of training a machine learning model to classify images while preserving the privacy of the images and their labels. This can be done by applying differential privacy to the training algorithm, such as stochastic gradient descent (SGD), which updates the model parameters using noisy gradients computed on batches of images.

Differential privacy image classification can help protect the privacy of sensitive images, such as faces, biometrics, or medical scans, while enabling useful applications, such as face recognition, biometric authentication, or medical diagnosis.

Examples of differential privacy in AI

Some examples of successful differential privacy in AI and their track record are:

1. **Apple**: Apple uses differential privacy to collect and analyze data from its users' devices, such as keyboard usage, emoji preferences, web browsing patterns, and health metrics, while protecting their privacy and identity. Apple claims that it does not see or store the raw data, but only aggregates the noisy data to improve its products and services, such as Siri, Safari, and HealthKit.
2. **Google**: Google uses differential privacy to collect and analyze data from its users' web and app activity, such as Chrome usage, YouTube views, and Maps searches, while protecting their privacy and choice. Google claims that it does not link or combine the noisy data with other data, but only uses it to improve its products and services, such as Chrome, YouTube, and Maps.
3. **Microsoft**: Microsoft uses differential privacy to collect and analyze data from its customers' devices, such as Windows usage, Office productivity, and Xbox gaming, while protecting their privacy and security. Microsoft claims that it does not access or store the raw data, but only uses the noisy data to improve its products and services, such as Windows, Office, and Xbox.

Video - Differential Privacy – So What

Differential Privacy Demos

Differential Privacy Demos - Differntial_Privacy.ipynb

Differential Privacy Synthetic Data Generator (SmartNoise Demo)

- **Purpose:** Show how **differential privacy (DP)** can be used to create **realistic synthetic datasets** that protect individuals' data while keeping statistical utility.
- **What this notebook does:**
  - **Upload a dataset (CSV)**
    - Any structured dataset (e.g. HR, finance, health records).
  - **Detect data types automatically**
    - Identifies categorical vs numerical columns.
  - **Train a DP synthesizer**
    - Uses Microsoft's **SmartNoise Synth** (patectgan / dpctgan) to learn data patterns under a controlled privacy budget (ε).
  - **Generate synthetic data**
    - Produces a new dataset with the same structure and statistical behaviour, but without exposing real individuals.
  - **Compare & download results**
    - Visualizes how well the synthetic data matches the real dataset.

Differential Privacy Demos - Differntial_Privacy.ipynb

Key Concept:

- **Differential Privacy adds mathematical "noise"** to protect individuals — Smaller ε = stronger privacy, lower accuracy Larger ε = weaker privacy, higher accuracy

- **Outcome:**
  - ✓ ✅ Synthetic data that mimics the original
  - ✓ 🔒 Privacy-preserving for safe data sharing and model testing

Differential Privacy Demos

Differential Privacy in Linear Regression

- **Purpose:** Show how adding *differential privacy (DP)* affects regression model accuracy.
- **Process Overview:**
  1. **Dataset:** Diabetes dataset (2 features).
  2. **Baseline Model:** Standard Linear Regression → $R^2 \approx$ **0.04**.
  3. **DP Model:** Uses diffprivlib.models.LinearRegression → Adds calibrated noise for **ε-differential privacy**.
  4. **Epsilon (ε):**
  5. Controls privacy–utility trade-off.
  6. **Low ε → high privacy, low accuracy.**
  7. **High ε → low privacy, high accuracy.**
  8. **Result:**
  9. As ε increases, R^2 gradually rises toward the non-private baseline.
  10. Visual plot shows R^2 vs ε (privacy–utility curve).
- **Key Insight:**
  - Differential privacy ensures data confidentiality by adding noise to model training — accuracy decreases slightly, but individual data protection is mathematically guaranteed.

Federated learning

What is federated learning?

Federated learning is a way to train AI models without anyone seeing or touching your data, offering a way to unlock information to feed new AI applications.

What is federated learning?

Federated learning (FL) is a machine learning approach that enables the training of a shared AI model using data from numerous decentralized edge devices or servers. This process occurs without the need to exchange the local data samples. Think of it as a collaborative learning process where individual participants contribute to a common goal without revealing their private information.

This contrasts sharply with traditional machine learning, which typically requires aggregating all data into a central repository for model training. While centralized approaches have driven significant AI advancements, they can raise concerns about data privacy, security, and compliance with regulations like GDPR. Federated learning offers a privacy-preserving alternative by keeping sensitive data localized on the user's device or within an organization's secure environment.

Federated learning versus machine learning

As mentioned above, the main difference between federated learning and traditional, centralized machine learning lies in where the data resides during the training process.

- **Traditional machine learning (centralized)**: Data is collected from various sources and brought together in one place, such as a cloud server or data center. The machine learning model is then trained directly on this consolidated dataset. This method can offer advantages like straightforward data access and simpler development, but it may also create significant privacy risks and potential vulnerabilities if the central data repository is compromised.
- **Federated learning (decentralized)**: Instead of moving data, the machine learning model is sent to the data, and participants (clients) train the model on their local data. Only the model updates—such as learned weights or gradients—are then sent back to a central server for aggregation. This process allows the global model to learn from diverse datasets without ever accessing the raw, sensitive information from any single participant.

While centralized machine learning is well established and often easier to implement, federated learning is gaining traction because it can inherently address data privacy concerns, reduce bandwidth requirements, and allow for model training on data that might otherwise be inaccessible due to regulations or confidentiality agreements.

How does federated learning work?

Federated learning works through an iterative process involving a central coordinator (typically a server) and multiple participating clients (devices or organizations). The general workflow can be broken down into these key steps:

1. Initial model distribution

The process begins with a central server initializing a global machine learning model. This model serves as the starting point for the collaborative training. The server then distributes this global model to a selected subset of participating client devices.

2. Local model training

Each selected client device receives the global model. Using its own local data, the client trains the model, updating its parameters based on the patterns and information present in that local dataset. Crucially, the raw data remains on the client device throughout this step, never being sent to the server.

3. Model update aggregation

After local training, each client sends its updated model parameters (for example, gradients or weights) back to the central server. These updates represent what the model learned from the local data, but they do not expose the data itself.

4. Global model update

The central server receives the model updates from multiple clients. It then aggregates these updates, often by averaging them (a common method being federated averaging, or FedAvg), to create a new, improved version of the global model. This aggregated model benefits from the collective learning across all participating clients.

5. Iterative refinement

The server then distributes this newly updated global model back to a new set of (or the same) clients for another round of local training. This cycle repeats multiple times, progressively refining the global model with each iteration until it reaches a desired level of accuracy or convergence.

What is federated learning?

Federated learning(FL) is a framework used to train a shared global model on data across distributed devices while the data does not leave the device at any point.

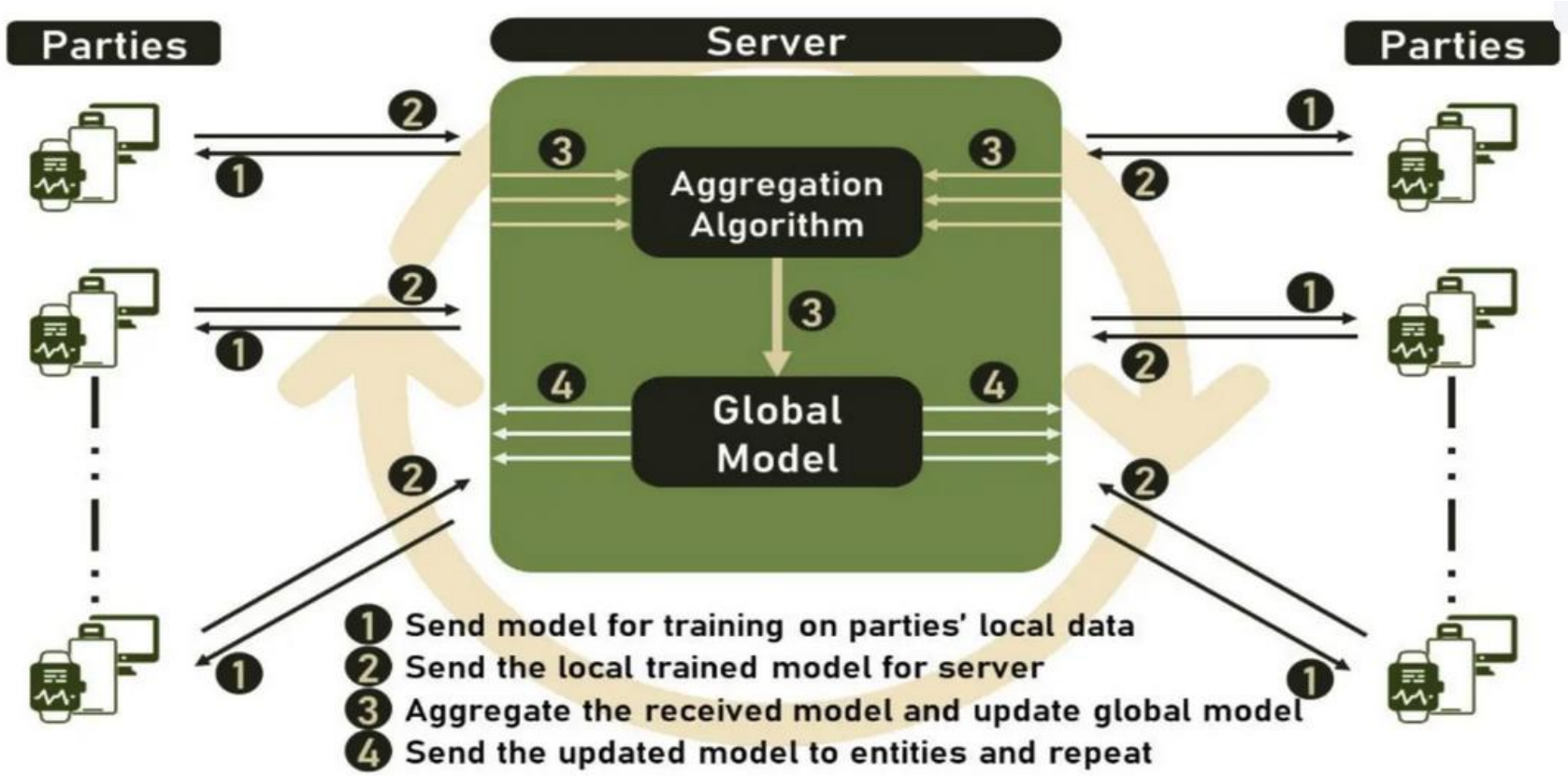
Central Server: the entity responsible for managing the connections between the entities in the FL environment and for aggregating the knowledge acquired by the FL clients;

Parties (Clients): all computing devices with data that can be used for training the global model, including but not limited to: personal computers, servers, smartphones, smartwatches, computerized sensor devices, and many more;

Communication Framework: consists of the tools and devices used to connect servers and parties and can vary between an internal network, an intranet, or even the Internet;

Aggregation Algorithm: the entity responsible for aggregating the knowledge obtained by the parties after training with their local data and using the aggregated knowledge to update the global model.

What is federated learning?



How Does Federated Learning Work?

Federated learning operates through a series of coordinated steps that enable collaborative model training while preserving data privacy. This process can be broken down into three key stages: **initialization**, **local training**, and **aggregation of updates**. Let's explore each stage in detail:

Initialization phase

The federated learning process begins with the initialization phase:

1. A central server develops an initial global model. This model serves as the starting point for the federated learning process.
2. The server distributes this global model to a selected group of participating client devices or servers. These clients could be smartphones, IoT devices, or local servers in different organizations.
3. Along with the model, the server sends instructions for training, including hyperparameters, the number of local epochs to perform, and any other relevant configuration details.
4. Clients receive the global model and prepare to train it on their local data.

How Does Federated Learning Work?

Local training

Once clients receive the global model, the local training phase begins:

1. Each client device trains the model using only its local data. This is a characteristic aspect of federated learning – the raw data never leaves the device.
2. The training process on each device is similar to traditional machine learning, which may involve forward passes, loss calculation, and backpropagation to update model parameters.
3. Clients perform a specified number of training epochs or iterations, as defined in the initialization phase.
4. After completing the local training, each client computes the difference between the updated model parameters and the original global model parameters. This difference represents the local update.

How Does Federated Learning Work?

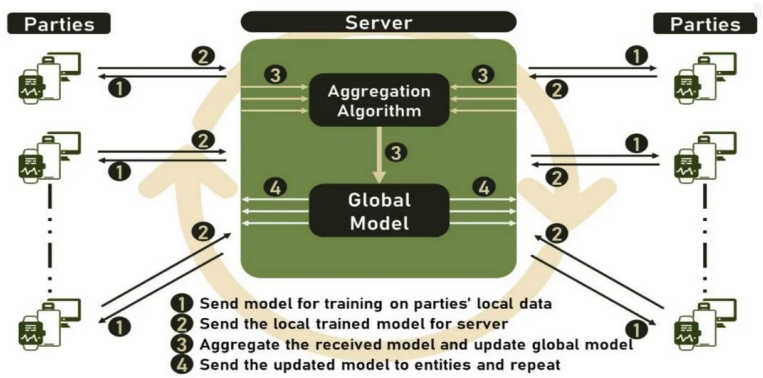
Aggregation of updates

The final stage involves aggregating the local updates to improve the global model:

1. Clients send their updated model parameters back to the central server. Importantly, *only the model updates are transmitted*, not the raw data or the fully trained local models.
2. The central server receives updates from multiple clients and aggregates them to create a new global model. This aggregation is typically done through a process called federated averaging, where the server computes a weighted average of all client updates.
3. To further enhance privacy, techniques like secure aggregation or differential privacy may be applied during this step. These methods add an extra layer of protection, making it virtually impossible to reverse-engineer individual contributions from the aggregated update.
4. The server updates the global model with the aggregated changes, creating an improved version that has learned from diverse data sources without directly accessing any local data.
5. This new global model is then distributed back to the clients, and the process repeats from the initialization phase for the next round of training.

Types of Federated Learning

Federated learning encompasses various approaches, each designed to address specific scenarios and challenges in distributed machine learning. While the core principle of training models on decentralized data remains constant, the implementation can vary. Let's explore four main types of federated learning:



Centralized federated learning

Centralized federated learning, also known as server-based federated learning, is the most common approach. The method we introduced above was characteristic of centralized federated learning. In centralized federated learning, a central server coordinates the entire learning process. The server then initiates training by distributing the global model to clients, who train the model locally and send updates back to the server. Finally, the server aggregates these updates to improve the global model.

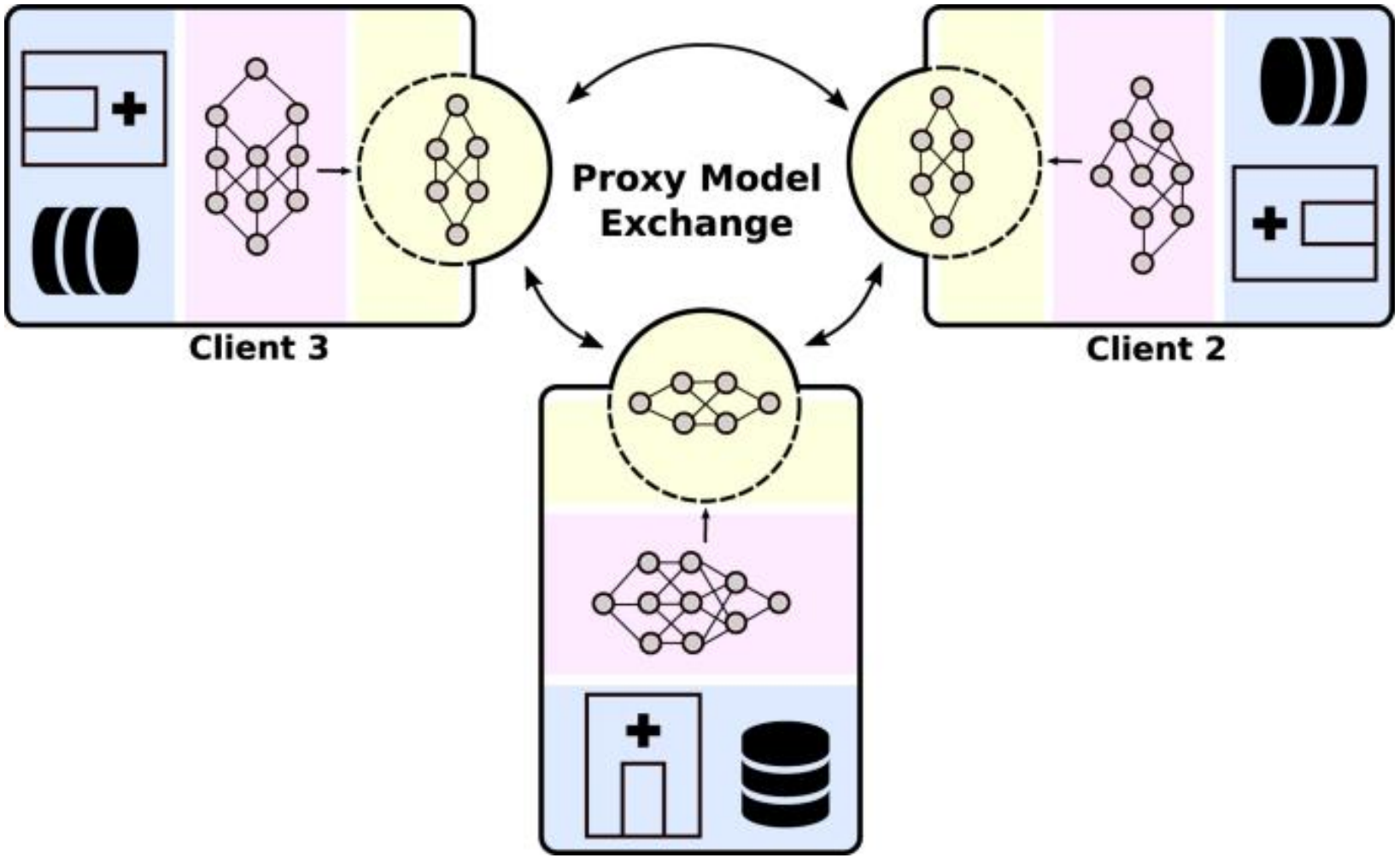
This approach is ideal for scenarios where a trusted central entity can manage the process, such as a tech company improving its services across user devices or a healthcare consortium coordinating research across multiple hospitals.

Decentralized federated learning

Decentralized federated learning eliminates the need for a central server. In decentralized federated learning, clients communicate directly with each other in a peer-to-peer network, and each client acts as both a learner and an aggregator. Models or updates are shared between clients, often using blockchain or other distributed ledger technologies, and the global model emerges from the collective interactions of all clients.

This approach is particularly useful in scenarios where no single trusted central authority exists or when enhanced privacy and resilience to single points of failure are required.

Decentralized federated learning

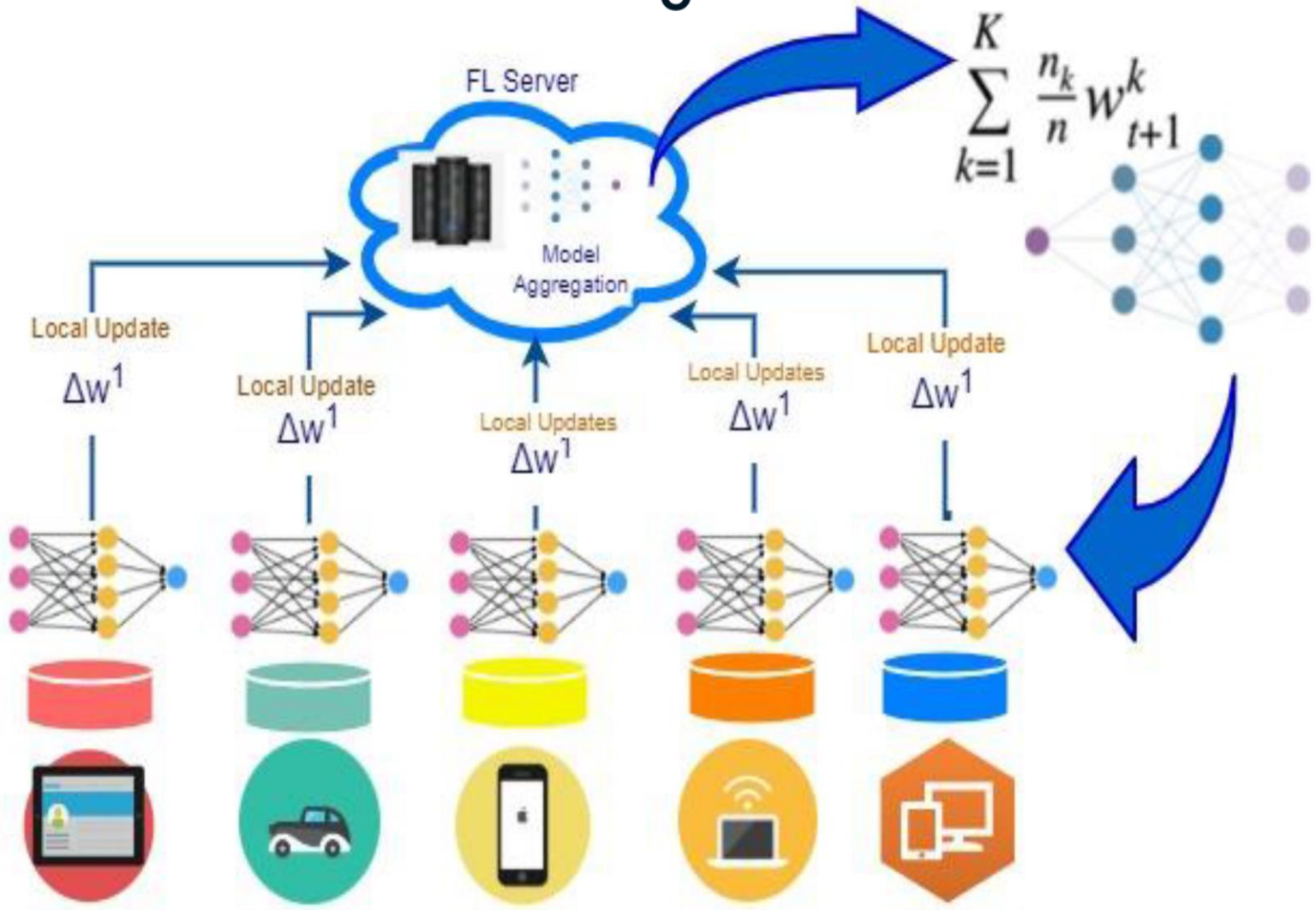


Heterogeneous federated learning

Heterogeneous federated learning addresses the challenges of training across diverse devices and data distributions because it accommodates devices with varying computational capabilities and resources. It does this by employing adaptive algorithms to handle varying data qualities and quantities across clients.

This type is useful in real-world applications where data and devices are inherently diverse, such as in IoT networks or when training models across different organizations.

Heterogeneous federated learning

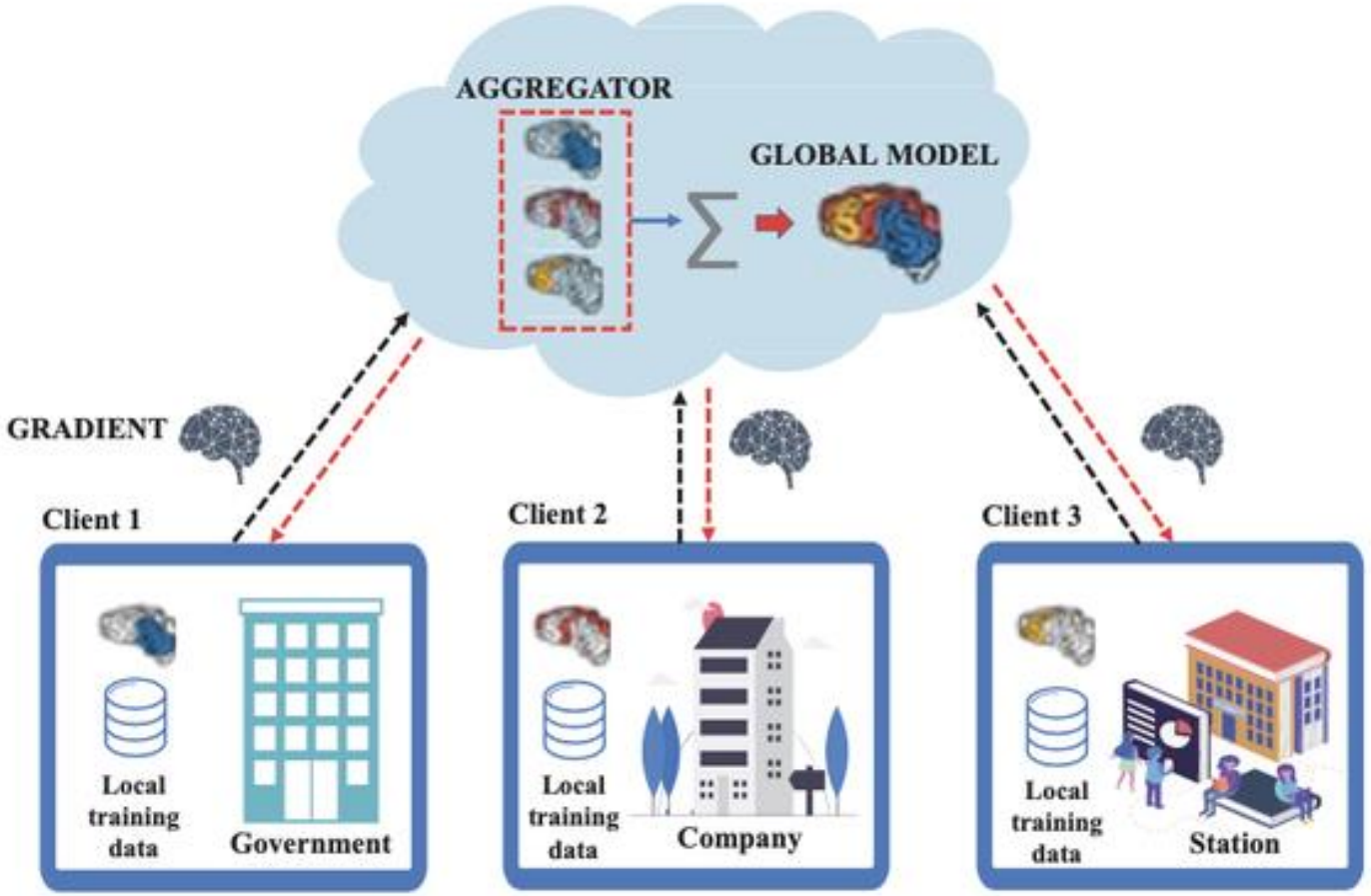


Cross-silo federated learning

Cross-silo federated learning focuses on collaboration between different organizations or data silos, It involves a small number of reliable participants, often organizations rather than individual devices. Participants typically have larger datasets and more stable connections compared to cross-device settings. It may involve complex legal and organizational agreements for data sharing and model ownership. It is often used in scenarios like collaborative research between institutions or inter-bank fraud detection systems.

This approach enables organizations to benefit from collective intelligence while maintaining control over their sensitive data.

Cross-silo federated learning



SUMMARY

What is federated learning?

Benefits of federated learning

Enhanced data privacy and security

This is arguably the most significant benefit. By keeping data localized on client devices, federated learning can drastically reduce the risk of sensitive information exposure during transmission or storage. This inherently enhances user privacy and helps organizations comply with stringent data protection regulations.

Access to diverse data

Federated learning allows models to learn from a wide array of real-world data sources that might otherwise be siloed or inaccessible. This diversity can lead to more robust, generalizable, and accurate models, as they're trained on a broader spectrum of user behaviors, conditions, or environments compared to models trained on a single, centralized dataset.

Collaborative model improvement

Federated learning enables organizations or individuals to collaborate on building and improving AI models without needing to share proprietary or sensitive data. This helps foster a more inclusive AI development ecosystem and allows for pooled intelligence from disparate sources.

Streamlined regulatory compliance

The inherent design of federated learning keeps data local, which can significantly aid in meeting complex data privacy regulations such as GDPR, CCPA, and HIPAA. By minimizing data movement and centralization, organizations can better ensure data residency requirements are met and reduce the compliance burden associated with handling sensitive personal or health information.

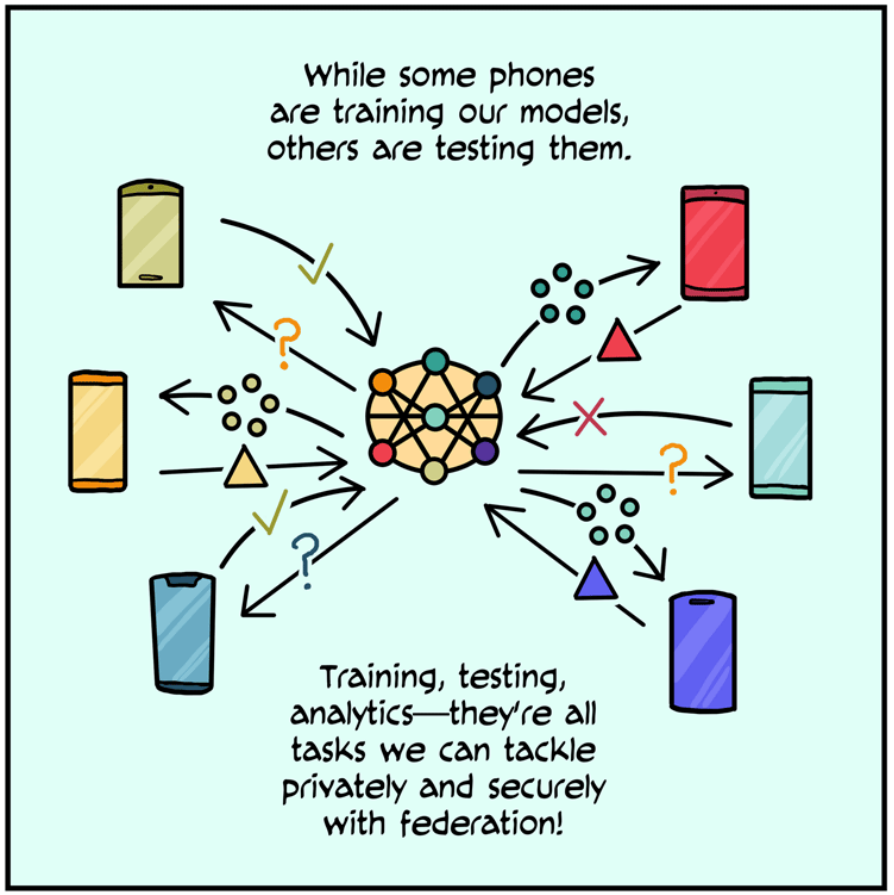
Benefits of federated learning

Reduced communication costs

Transmitting model updates (which are typically smaller than raw datasets) is often more bandwidth-efficient and less costly than transferring massive amounts of raw data to a central server, especially in scenarios involving many edge devices or geographically dispersed locations.

Upholding data sovereignty

This approach respects data ownership and control. Participating organizations or individuals retain full authority over their data assets. Even when contributing to a collective model, the raw data remains securely within its original environment, empowering data governance and maintaining trust between collaborators.



Technical Challenges and Advancements

- Communication Overhead: Frequent communication of model updates can be resource-intensive, especially for geographically distributed clients.
- Non-IID Data: Clients might have data with different distributions (non-IID), hindering the effectiveness of the global model.
- Privacy Leakage: Even with model updates, there's a risk of inferring sensitive information through reconstruction attacks.

Advancements

- Efficient Aggregation Protocols: Techniques like selective aggregation or model averaging can reduce communication overhead.
- Federated Transfer Learning: Pre-training a base model on a diverse dataset helps address non-IID data issues.
- Differential Privacy with Secure Aggregation: Adding controlled noise to model updates combined with secure aggregation methods can further enhance privacy protection.

Applications of Federated Learning

Federated learning is changing how industries approach machine learning, especially in scenarios where data privacy and security are paramount. By enabling collaborative learning without centralizing sensitive data, federated learning is finding applications in different sectors.

Healthcare

Federated learning is transforming medical research and patient care by allowing hospitals and research institutions to collaborate without compromising patient privacy:

- **Cancer Research**: The MELLODDY project, involving ten pharmaceutical companies, uses federated learning to improve drug discovery for cancer treatments without sharing proprietary data.
- **Predictive Healthcare**: Hospitals can collaboratively train models to predict patient outcomes, readmission risks, or rare disease diagnoses using data from multiple institutions.
- **Medical Imaging**: Federated learning enables the development of more robust AI models for interpreting X-rays, MRIs, and CT scans by learning from diverse datasets across different healthcare providers.
- **Pandemic Response**: During the COVID-19 pandemic, federated learning facilitated rapid collaboration between hospitals worldwide to develop prediction models for patient outcomes and resource allocation.

Finance

Financial institutions are leveraging federated learning to enhance their services while maintaining strict data privacy and regulatory compliance:

- **Fraud Detection**: Banks collaborate to train more effective fraud detection models without sharing sensitive transaction data, as demonstrated by projects like the one led by WeBank in China.
- **Credit Scoring**: Lenders can develop more accurate credit risk assessment models by learning from diverse customer bases across multiple institutions without centralizing personal financial data.
- **Anti-Money Laundering (AML)**: Financial institutions use federated learning to improve AML detection systems by collaboratively training on patterns from various banks without exposing individual transaction details.
- **Personalized Financial Services**: Banks can offer product recommendations and investment advice based on models trained across diverse customer bases while keeping individual customer data private.

Federated Learning

Building better products with on-device data and privacy by default

An online comic from Google AI

CREST CISA Certified Information Systems Auditor. An ISACA® Certification bsi ISO/IEC 27001 Information Security Management CERTIFIED bsi ISO 9001 Quality Management Systems CERTIFIED Crown Commercial Service

Assentian Limited

Federated learning

DEMO

Federated learning

- **The Scenario**
  - Imagine a breast cancer study with two key people:
- **Owen (The Data Owner):** He works at a Cancer Research Centre and protects a **private patient dataset**. Due to legal and privacy reasons, he **cannot** share this data with anyone.
- **Rachel (The Data Scientist):** She's a researcher who wants to build an AI model to analyze that data.
- **The Problem**
  - How can Rachel analyse the data if she can't see it?
- **The Solution**
  - Instead of sending the **data** to Rachel, Rachel sends her **analysis code** (her "plan") to Owen's secure server.
  - Owen **reviews** her code to make sure it's safe.
  - The code runs **inside Owen's secure server**.
  - Rachel **only gets the final result** (e.g., "the model is 95% accurate"), not the data itself.
  - **The private data never leaves the server.**

Federated learning

The Secure Workflow – How It Works: The notebook walks through this 4-step process:

- **1. Setup (Owen's Job)**
  - Owen uploads the **real, private data** to his secure server (called a "Datasite").
  - He also creates and uploads **fake "mock" data**. This fake data has the same structure (columns, etc.) as the real data but is scrambled and contains no private information.
  - He creates a user account for Rachel.
- **2. Propose (Rachel's Job)**
  - Rachel logs in. She can **only** see and download the *mock* data.
  - She uses this fake data to write and test her AI analysis code until it works.
  - She submits her finished code to Owen as a "project" for approval.

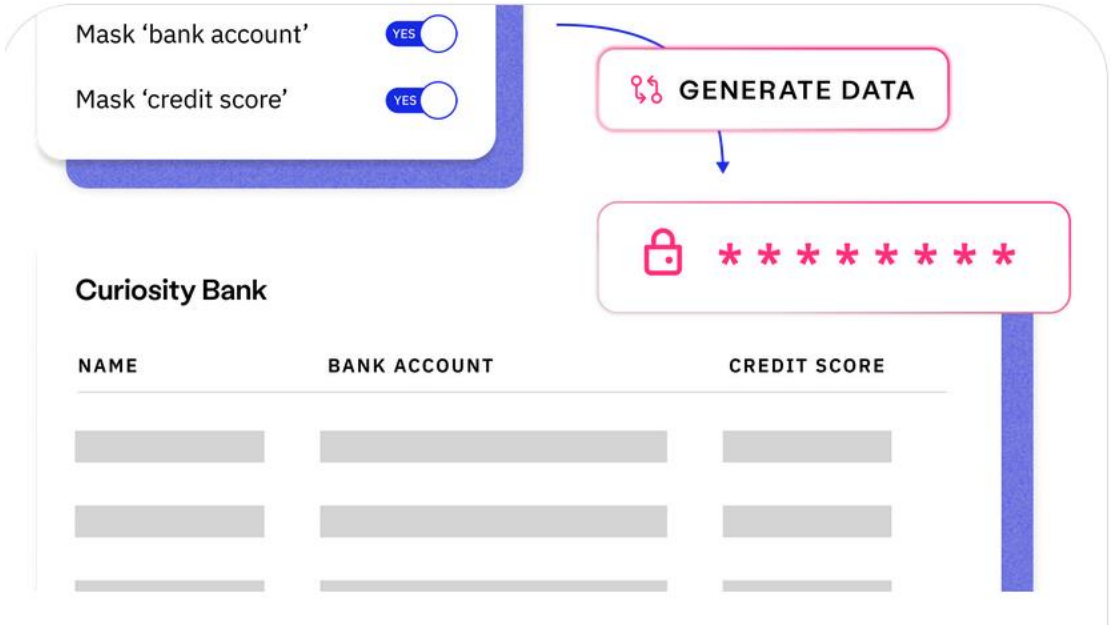
Federated learning

- **3. Review (Owen's Job)**
  - Owen gets an alert for Rachel's project.
  - He reads her code to make sure it's safe (e.g., it only calculates a result and doesn't try to copy the data).
  - He tests the code, sees it's safe, and **approves** her request.
- **4. Get Result (Rachel's Job)**
  - Rachel sees her code is now approved.
  - She can now "run" her code. It runs remotely on Owen's server using the *real* data.
  - She gets **only the final result** back. She has successfully completed her research without ever seeing a single patient's record.

Properties of privacy-preserving synthetic data

What is synthetic data?

Synthetic data is non-human-created data that mimics real-world data. It is created by computing algorithms and simulations based on generative artificial intelligence technologies. A synthetic data set has the same mathematical properties as the actual data it is based on, but it does not contain any of the same information. Organizations use synthetic data for research, testing, new development, and machine learning research. Recent innovations in AI have made synthetic data generation efficient and fast but have also increased its importance in data regulatory concerns.



CREST
CISA Certified Information Systems Auditor. An ISACA® Certification
bsi ISO/IEC 27001 Information Security Management CERTIFIED
bsi ISO 9001 Quality Management Systems CERTIFIED
Crown Commercial Service

Assentian Limited

What are the benefits of synthetic data?

Unlimited data generation

You can produce synthetic data on demand and at an almost unlimited scale. Synthetic data generation tools are a cost-effective way of getting more data. They can also pre-label (categorise or mark) the data they generate for machine learning use cases. You get access to structured and labeled data without going through the process of transforming raw data from scratch. You can also add synthetic data to the total volume of data that you have, yielding more training data for analysis.

Privacy protection

Fields like healthcare, finance, and the legal sector have many privacy, copyright, and compliance regulations to protect sensitive data. However, they must use data for analytics and research—often having to outsource data to third parties for maximum utilization. Instead of personal data, they can use synthetic data to serve the same purpose as these private datasets. They create similar data that shows the same statistically relevant information without exposing private or sensitive data. Consider medical research creating synthetic data from a live data set— the synthetic data maintains the same percentage of biological characteristics and genetic markers as the original data set, but all names, addresses, and other personal patient information is fake.

What are the benefits of synthetic data?

Bias reduction

You can use synthetic data to reduce bias in AI training models. As large models typically train on publicly available data, there can be bias in the text. Researchers can use synthetic data to provide a contrast to any biased language or information that AI models collect. For example, if certain opinion-based content is favoring a particular group, you can create synthetic data to balance out the overall dataset.

What are the types of synthetic data?

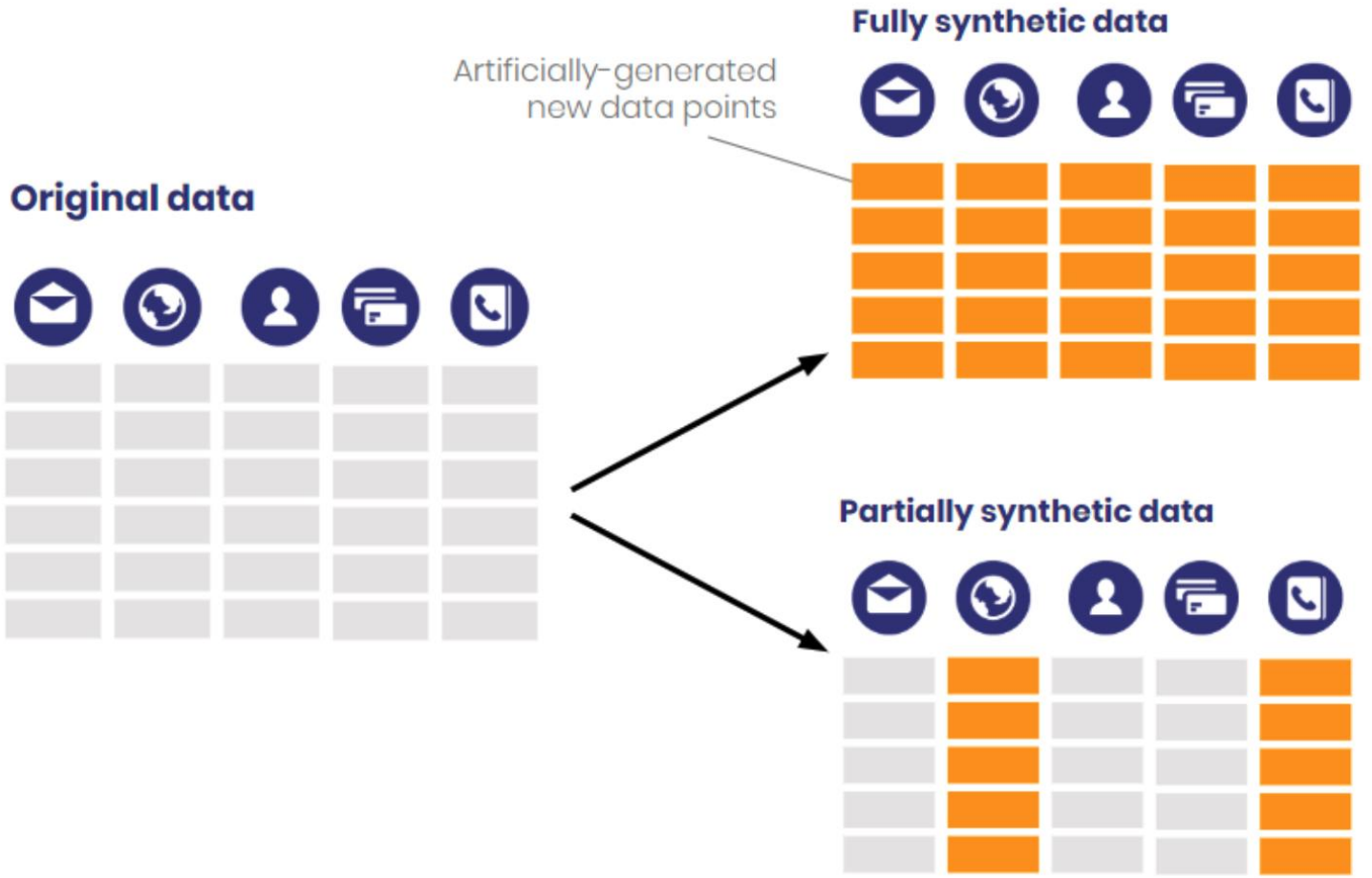
Partial synthetic data

Partially synthetic data replaces a small portion of a real dataset with synthetic information. You can use it to protect sensitive parts of a dataset. For example, if you need to analyze customer-specific data, you can synthesize attributes like name, contact details, and other real-world information that someone could trace back to a specific person.

Full synthetic data

Full synthetic data is where you completely generate new data. A fully synthetic dataset will not contain any real-world data. However, it will use the same relationships, plot distributions, and statistical properties as real data. While this data doesn't come from actual recorded data, it allows you to make the same conclusions.

You can use fully synthetic data when testing machine learning models. It is useful when you want to test or create new models but dont have sufficient real-world training data for improved ML accuracy.


Fully synthetic data
Artificially-generated
new data points
Original data
Partially synthetic data

How is synthetic data generated?

Synthetic data generation involves the use of computational methods and simulations to create data. The result mimics the statistical properties of real-world data, but does not contain actual real-world observations. This generated data can take various forms, including text, numbers, tables, or more complex types like images and videos. There are three main approaches to generating synthetic data, each offering different levels of data accuracy and types.

Statistical distribution

In this approach, real data is first analyzed to identify its underlying statistical distributions, such as normal, exponential, or chi-square distributions. Data scientists then generate synthetic samples from these identified distributions to create a dataset that statistically resembles the original.

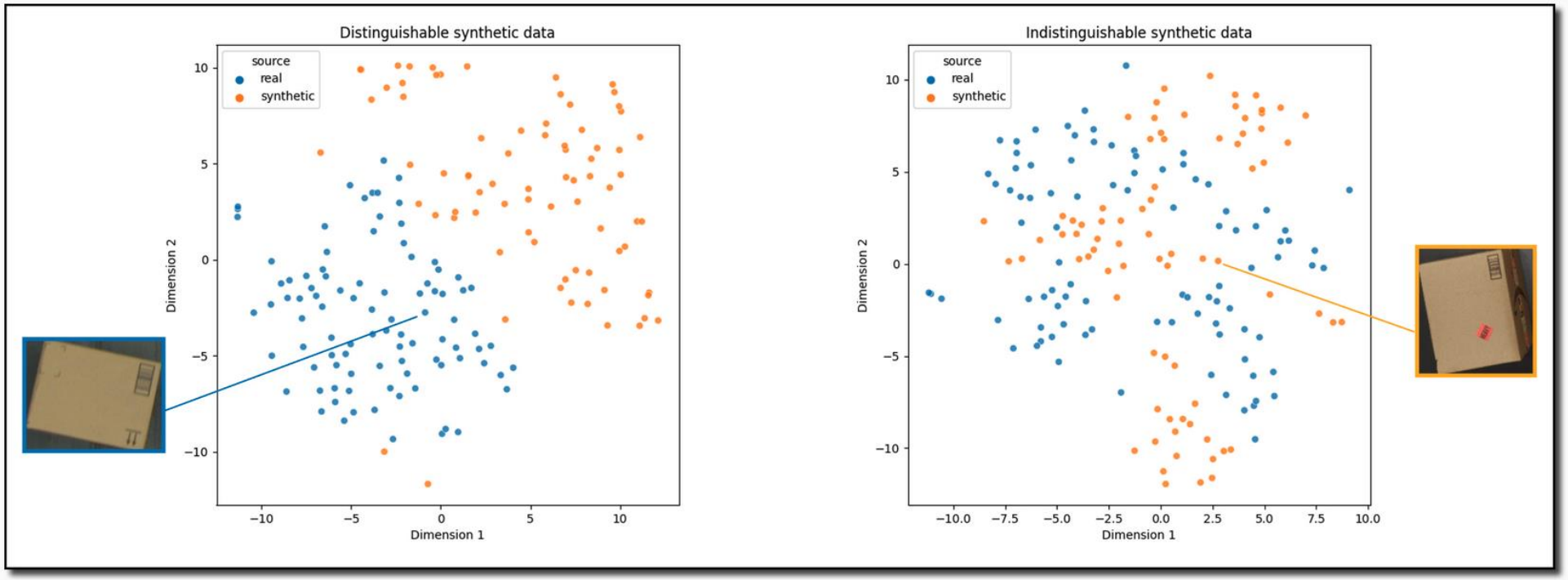
Model-based

In this approach, a machine learning model is trained to understand and replicate the characteristics of the real data. Once the model has been trained, it can generate artificial data that follows the same statistical distribution as the real data. This approach is particularly useful for creating hybrid datasets, which combine the statistical properties of real data with additional synthetic elements.

How is synthetic data generated?

Deep learning methods

Advanced techniques like Generative adversarial networks (GANs), variational autoencoders (VAEs), and others can be employed to generate synthetic data. These methods are often used for more complex data types—like images or time-series data—and can produce high-quality synthetic datasets.



How is synthetic data generated?

What is a GAN?

A generative adversarial network (GAN) is a deep learning architecture. It trains two neural networks to compete against each other to generate more authentic new data from a given training dataset. For instance, you can generate new images from an existing image database or original music from a database of songs. A GAN is called *adversarial* because it trains two different networks and pits them against each other. One network generates new data by taking an input data sample and modifying it as much as possible. The other network tries to predict whether the generated data output belongs in the original dataset. In other words, the predicting network determines whether the generated data is fake or real. The system generates newer, improved versions of fake data values until the predicting network can no longer distinguish fake from original.

How does a generative adversarial network work?

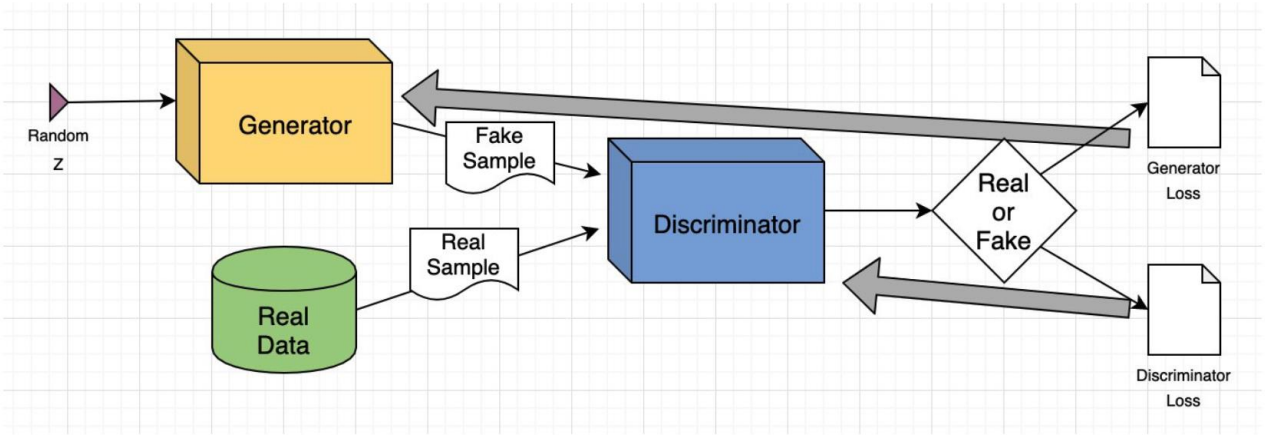
A generative adversarial network system comprises two deep neural networks—the *generator network* and the *discriminator network*. Both networks train in an adversarial game, where one tries to generate new data and the other attempts to predict if the output is fake or real data.

Technically, the GAN works as follows. A complex mathematical equation forms the basis of the entire computing process, but this is a simplistic overview:

1. The generator neural network analyzes the training set and identifies data attributes
2. The discriminator neural network also analyzes the initial training data and distinguishes between the attributes independently
3. The generator modifies some data attributes by adding noise (or random changes) to certain attributes
4. The generator passes the modified data to the discriminator
5. The discriminator calculates the probability that the generated output belongs to the original dataset
6. The discriminator gives some guidance to the generator to reduce the noise vector randomization in the next cycle

The generator attempts to maximize the probability of mistake by the discriminator, but the discriminator attempts to minimize the probability of error. In training iterations, both the generator and discriminator evolve and confront each other continuously until they reach an equilibrium state. In the equilibrium state, the discriminator can no longer recognize synthesized data. At this point, the training process is over.

How does a generative adversarial network work?



GAN training example

Let's contextualize the above with an example of the GAN model in image-to-image translation.

Consider that the input image is a human face that the GAN attempts to modify. For example, the attributes can be the shapes of eyes or ears. Let's say the generator changes the real images by adding sunglasses to them. The discriminator receives a set of images, some of real people with sunglasses and some generated images that were modified to include sunglasses.

If the discriminator can differentiate between fake and real, the generator updates its parameters to generate even better fake images. If the generator produces images that fool the discriminator, the discriminator updates its parameters. Competition improves both networks until equilibrium is reached.

What are the challenges in synthetic data generation?

Quality control

Data quality is vital in statistics and analytics. Before you incorporate synthetic data into learning models, you must check that it is accurate and has a minimum level of data quality. However, ensuring that no-one can trace synthetic data points back to real information may require a reduction in accuracy. A trade-off in privacy and accuracy could impact quality.

You can perform manual checks of synthetic data before you use it, which can help to overcome this issue. However, manually checking can become time-consuming if you need to generate lots of synthetic data.

Technical challenges

Creating synthetic data is difficult—you must understand techniques, rules, and current methods to ensure its accuracy and utility. You need high expertise in this field before you'll be generating any useful synthetic data.

No matter how much expertise you have on your side, it is challenging to generate synthetic data as a perfect imitation of its real-world counterpart. For instance, real-world data often includes outliers and anomalies that synthetic data generation algorithms can rarely recreate.

Synthetic Data from Upload Configure Metadata

Generate Synthetic Data from a File

Here you can upload files either to generate metadata that can be used to create synthetic data later or create synthetic data directly from a dataset.

Generate Synthetic Data

Upload a dataset (csv) or metadata (json) and generate synthetic data from it automatically.

Size

1000

Browse... No file selected.

Generate Data

OR

Generate Metadata

Upload a dataset (csv) and receive a metadata file describing your dataset that can be used to generate data later.

Browse... No file selected.

Generate Metadata

CREST CISA Certified Information Systems Auditor. An ISACA® Certification bsi ISO/IEC 27001 Information Security Management CERTIFIED bsi ISO 9001 Quality Management Systems CERTIFIED Crown Commercial Service

Assentian Limited

Synthetic Data from Upload | Configure Metadata

Configure Your Own Metadata

Here you can configure your own metadata and generate a dataset based on it.

Options

Submit or Export

Submit your metadata to generate a dataset or export it as a json file.

Number of Rows to Generate

1000

Submit | Export

Columns

Column Settings

Here you can configure what data you expect to see in each generated column.

Add or Overwrite Column

Here you can configure a column to add. The description below explains what column you will add with your current settings.

Type column name here... | numerical-uniform | Add +

Description

Please select a column name to add it. This column is a numerical column with a uniform distribution. This column type has a configurable lower bound and upper bound. Values generated for this column will not go lower or higher than their respective bounds.

Correlations

Correlation Settings

CREST | CISA Certified Information Systems Auditor. An ISACA® Certification | bsi ISO/IEC 27001 Information Security Management CERTIFIED | bsi ISO 9001 Quality Management Systems CERTIFIED | Crown Commercial Service

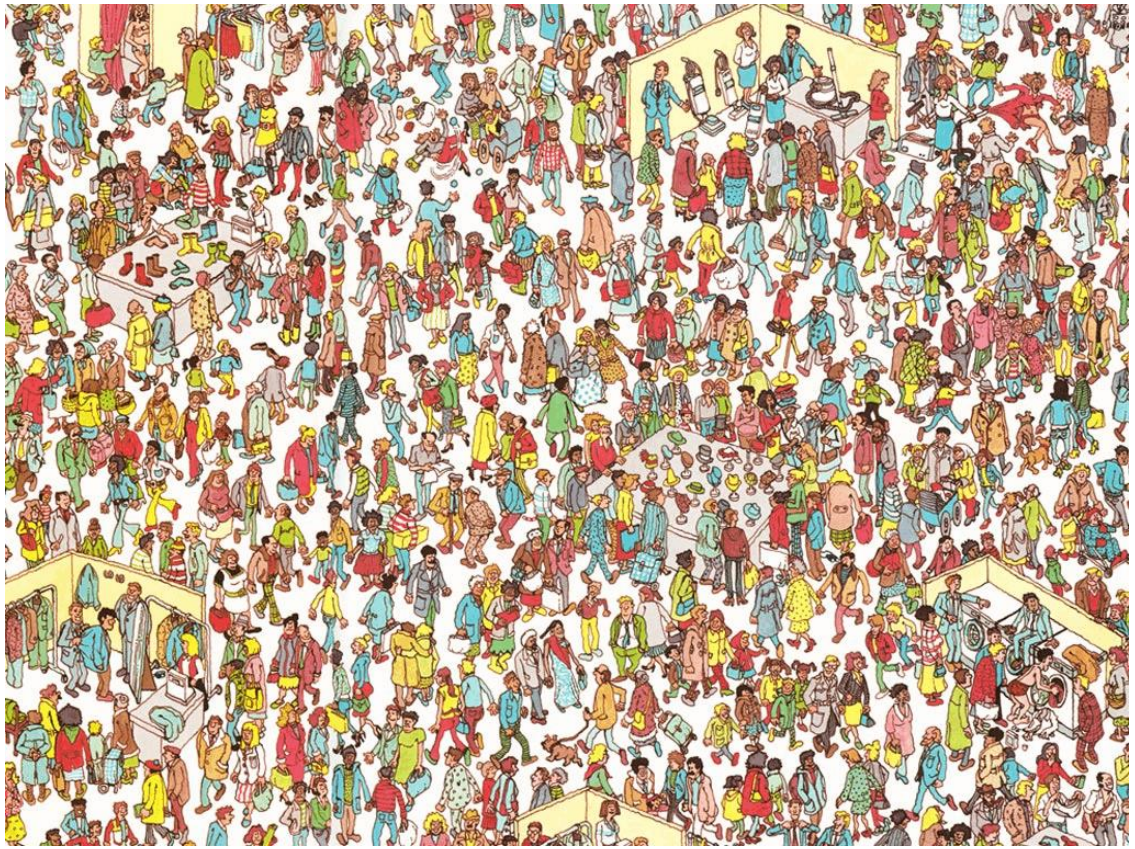
Assentian Limited

Synthetic Data Generator Demo

Zero-Knowledge Proofs

Zero-knowledge proofs (ZKPs) are a type of cryptographic method that allows one party (the prover) to prove to another party (the verifier) that they possess a certain piece of information without actually revealing that information. In other words, the prover can prove to the verifier that they know something without telling the verifier what that something is.

- **Interactive proofs:** These are the most basic type and involve a prover and a verifier who interact with each other to prove the prover's knowledge.
- **Non-interactive proofs:** These are more complex than interactive proofs and involve a prover who creates a proof without interacting with the verifier. Non-interactive proofs are often used when the prover and verifier cannot communicate directly or when the prover wants to keep their identity private.



- **Cryptocurrency transactions:** They can be used to verify the authenticity of cryptocurrency transactions without revealing the details of the transaction itself. This can help protect the parties' privacy in the transaction.
- **Authentication:** As mentioned earlier, ZKPs can be used to prove that a user knows a particular secret (such as a password) to access a system or service.

You take a massive piece of paper to cover up the entire image, showing your friend the image of Wally through a cutout. You can prove that you really know Wally's location, yet your friend will not gain knowledge of where Wally is since the exact coordinates of Wally relative to the image would still be unknown to him.

Zero-Knowledge Machine Learning (ZKML): Enabling Trustless AI

ZKML is an emerging field that applies **Zero-Knowledge Proofs (ZKPs)** to machine learning models. The goal is to allow AI models to perform computations without revealing their inputs, outputs, or internal logic. This ensures **privacy, verifiability, and decentralization**, making it possible to deploy AI in sensitive environments without data leaks.

How ZKML Works

At its core, ZKML combines **SNARKs (Succinct Non-Interactive Argument of Knowledge)** or **STARKs (Scalable Transparent Argument of Knowledge)** with AI models. The process involves:

1. **Encoding ML Models into Cryptographic Proofs:** AI computations are transformed into verifiable mathematical statements.
2. **Generating Zero-Knowledge Proofs:** Instead of revealing raw data, the system generates cryptographic proofs that verify the correctness of the AI's output.
3. **Trustless Verification:** A verifier (such as a blockchain node or decentralized entity) confirms the AI's output without accessing the original data or model weights.

- **SNARKs (Succinct Non-Interactive Argument of Knowledge**), are a type of cryptographic proof that allow a prover to convince a verifier that a statement is true without revealing any information beyond the validity of the statement itself
- These proofs are "succinct," meaning they have small proof sizes and fast verification times that do not depend on the complexity of the statement.
- They are "non-interactive," as the prover sends a single message to the verifier, and "arguments of knowledge" to ensure the prover genuinely possesses the secret information (called a witness) they are proving with.

- **STARKs, or Scalable Transparent ARguments of Knowledge**, are a type of cryptographic proof that allows one party to prove the validity of a computation without revealing the underlying data.
- They are "scalable" because proof generation and verification are efficient, even for large datasets, and "transparent" because they don't require a trusted setup, relying instead on public randomness.
- STARKs are also resistant to quantum computers because they use hash-based cryptography

Group Exercise: The "Privacy-by-Design" AI Clinic

This role-playing exercise challenges groups to apply various PETs to real-world AI scenarios, balancing data utility with privacy protection.

Objective: To understand how different PETs (e.g., Federated Learning, Homomorphic Encryption, Differential Privacy, Synthetic Data) can be used to mitigate privacy risks throughout the AI lifecycle.

Phase 1 – Set Up

Form Groups: Divide participants into small groups.

Assign Roles (within each group):

Data Ethicist: Focuses on individual rights, fairness, and regulatory compliance.
AI/ML Engineer: Focuses on the technical feasibility, model performance, and data utility.
Business Stakeholder: Focuses on project viability, data sharing opportunities, and commercial interests.
Project Manager: Facilitates discussion, manages time, and presents the group's solution.

The following PETs should form part of the Options

Homomorphic Encryption
Differential Privacy
Federated Learning
Synthetic Data Generation

Phase 2 – Scenario Analysis & Solution Design

Each group is given one of the following scenarios (or all groups work on the same one for comparison):

Scenario: Cross-Bank Fraud Detection

The Goal: A consortium of major banks wants to build a shared AI model to detect complex fraud patterns across their collective customer transaction data.
The Problem: Banks cannot share raw customer transaction data due to strict privacy regulations, commercial sensitivity, and trust issues.
The Task: Design a solution using one or a combination of PETs that enables the collaborative AI model training while ensuring no raw customer data is exposed to other banks or the central model owner.

Scenario: Personalized Healthcare Diagnostics

The Goal: A hospital wants to use an AI model, trained on diverse patient health records (including sensitive genetic data and medical images) from various clinics, to improve early disease diagnosis.
The Problem: Patient data is highly sensitive and protected by strict HIPAA-like regulations. Sharing raw data for training is a major privacy risk and compliance challenge.
The Task: Propose an AI architecture and select appropriate PETs to train a high-performing model without compromising individual patient privacy or data confidentiality.

Phase 2 – Scenario Analysis & Solution Design

Group Activity:

Groups analyze their scenario.

Roles drive the discussion (Ethicist ensures compliance, Engineer ensures performance, Business Stakeholder ensures collaboration is possible, PM keeps them on track).

They select which PETs are most suitable for different stages of the AI lifecycle (data collection, training, deployment).

They discuss the trade-offs: How might the chosen PETs impact data utility, performance, or implementation cost?

Phase 3: Presentation and Debrief

Present Solutions: Each group (via their Project Manager) presents their proposed PET solution, justifying their choices and trade-offs.

Peer Review/Q&A: Other groups and the facilitator ask questions, challenging the solution's feasibility, privacy guarantees, or scalability.

Facilitator-led Debrief:

- Highlight the strengths and weaknesses of different approaches.
- Emphasize that no single PET is a "magic bullet" and often a combination is needed.
- Reinforce the concept of "privacy by design" as a process of integrating privacy into the core architecture, not an afterthought.

The AI Lifecycle

THE AI LIFE CYCLE

Organisations are responsible for managing risks and harms throughout the AI lifecycle by implementing effective AI governance controls.

CHALLENGES

Data

The quality of training, testing, validation and operational datasets used to develop and operate AI systems can generate risks.

Privacy and data protection

The inherent dependency of AI on data can conflict with fundamental privacy principles like data minimization and purpose specification.

Transparency, explainability and interpretability

Users and developers both struggle to explain or interpret the inner workings and outputs of AI systems.

Bias, discrimination and fairness

Bias can encode into AI throughout the system life cycle through the data, algorithms or humans involved in developing and deploying the system.

Security and robustness

Compromised security of AI systems could lead to a range of harms, from incorrect outputs to physical harm.

AI safety

Safety risks include alignment, security, malicious use and rogue behavior risks.

Copyright

Training data for generative AI may include copyrighted content, raising issues related to infringement and fair use.

PRACTICAL APPROACHES

- DATA GOVERNANCE
- DATA MANAGEMENT PLANS
- DATA LABELS
- CONFORMITY ASSESSMENTS
- INVENTORIES
- PRIVACY BY DESIGN
- RISK ASSESSMENTS
- MODEL AND SYSTEM CARDS
- OPEN-SOURCE AI
- WATERMARKING
- PUBLIC-FACING INTERNAL AI ETHICS POLICIES
- BIAS TESTING
- IMPACT ASSESSMENTS
- HUMAN IN THE LOOP
- RED TEAMING
- SECURE DATA SHARING PRACTICES SUCH AS DIFFERENTIAL PRIVACY
- POST-MARKET MONITORING
- PROMPT ENGINEERING
- REPORTS AND COMPLAINTS
- SAFETY BY DESIGN
- INTERNAL AI SAFETY POLICIES
- TESTING AND EVALUATION
- OPT OUTS
- LIABILITY CONSIDERATIONS
- TECHNICAL GUARDRAILS SUCH AS CONTENT FILTERS, ABUSE DETECTION AND CLASSIFIERS

CROSS-CUTTING APPROACHES: RISK MANAGEMENT | TARGET OPERATING MODELS | POLICY AND PROCEDURE | COMPLIANCE ASSESSMENTS | TRAINING AND AWARENESS

CREST

CISA Certified Information Systems Auditor. An ISACA® Certification

bsi ISO/IEC 27001 Information Security Management CERTIFIED

bsi ISO 9001 Quality Management Systems CERTIFIED

Crown Commercial Service

Assentian Limited

Homomorphic Encryption: How It Works

HE in the AI Lifecycle – Data Collection & Sharing

- Sensitive/raw data encrypted before leaving data owner
- Seamless cross-organization and cross-border data sharing
- Enables collaborative data pooling for richer, bias-reduced AI models
- PETs like HE overcome data sovereignty barriers in global AI teams

With HE, raw data never leaves its source unprotected. HE-encoded data is safe for sharing even with external AI vendors or between jurisdictions with strict data localization rules. Multiple organizations can combine encrypted datasets for joint model training or analysis, without seeing each other's underlying data. This expands AI's data universe in a privacy-compliant way and reduces bias by accessing more representative data

HE in the AI Lifecycle – HE in AI Model Training

- Model parameters can be learned from encrypted inputs
- Prevents data leakage, even to infrastructure/providers
- Used in privacy-preserving federated and collaborative AI
  - Technical note: FHE for full training is still resource-intensive; more common for partial steps or secure aggregation

HE enables portions of the model training pipeline—most notably secure aggregations or partial computations—to be performed on encrypted input, preventing leaks even in untrusted environments. While full-scale training on encrypted data is computationally demanding, ongoing research brings this closer to reality. For now, hybrid approaches using HE in combination with other PETs are widely adopted for privacy-focused collaboration

HE in the AI Lifecycle – HE for Secure AI Inference and Deployment

- End-users send encrypted queries to AI/ML models (cloud or SaaS)
- Models process ciphertext, returning encrypted predictions to user
- No exposure of proprietary data, sensitive queries, or outputs
- Example: Medical image classification, financial risk scoring, with no plaintext data ever exposed

In inference scenarios, organizations (like hospitals or banks) encrypt sensitive input (e.g., patient images, account data) and outsource computation to a powerful, potentially untrusted AI model. Only the requester can decrypt results, ensuring that both input and predictions remain private—even from service providers. This unlocks the power of cloud AI without the privacy risks of sending data in plaintext

HE in the AI Lifecycle – HE: Compliance, Trust, and Risk Reduction in AI

- Aligns with GDPR, HIPAA, and other data protection mandates
- Reduces risk of data breaches and misuse
- Strengthens trust in cloud and third-party AI deployments
- Enables auditability (encrypted logs, encrypted model operations)

Homomorphic encryption directly addresses core requirements of data protection law by ensuring "data minimization" and "privacy by design." It is highly effective at preventing data exposure from breaches, insider threats, or supply chain compromise. Deploying HE can act as a trust signal, assuring partners and the public of robust privacy-preserving AI practices, and supports strong audit trails for compliance investigations

Multi-Party Computation: Collaborative Analysis Without Sharing Data

Multi-Party Computation: Collaborative Analysis Without Sharing Data

MPC in the AI Lifecycle – MPC in Data Collection & Sharing

- Pool distributed data from multiple parties for AI model training
- Data silos overcome with privacy-preserving collaboration
- Common methods: secret sharing, private set intersection, garbled circuits
- Enables secure data acquisition, dataset matching, and quality checks with no leakage

At the data collection phase, MPC enables organizations to securely combine datasets from multiple sources. Techniques like secret sharing and private set intersection let parties compute joint statistics, find matches, or check dataset compatibility—all without giving up original data. This unlocks collaborative AI opportunities while preventing exposure of sensitive inputs (e.g., hospitals can pool patient records for research, or banks can share fraud data)

MPC in the AI Lifecycle – MPC in AI Model Training and Analytics

- Joint model training and analytics on combined data
- No party learns others' data; only aggregate model parameters exposed
- Used for privacy-preserving federated learning, cross-border analysis, secure benchmarking
- Example: Multiple hospitals train a cancer detection model without sharing patient records

During training, MPC lets participants build models or analyze combined data securely. Each party's data is kept secret, but the collective benefit—accurate, less biased AI—is realized. MPC is foundational for privacy-preserving federated learning, collaborative research, and cloud analytics where data sovereignty or competition is a concern. Real world: Hospitals contribute encrypted patient data to train disease models that none could train alone

MPC in the AI Lifecycle – MPC in AI Inference and Deployment

- Secure prediction generation with distributed input
- Enables joint risk analysis, fraud detection, decision support across parties
- No exposure of sensitive queries, models, or outputs
- Applied in consortium lending, insurance risk, federated healthcare diagnostics

In the inference phase, MPC ensures secure calculation of predictions when data or models are distributed. For example, a consortium of banks can jointly assess loan risk, or insurers can share analytics for claims, all without revealing underlying customer or proprietary data. MPC guarantees results while locking down every party's inputs throughout deployment and routine operations

MPC in the AI Lifecycle – MPC – Compliance, Trust, and Regulation

- Endorsed by EU Data Protection Board and ENISA for international transfers
- Meets “privacy by design,” “data minimization” requirements of GDPR, HIPAA, etc.
- Reduces risk of leaks, misuse, and regulatory non-compliance
- Demonstrates responsible, privacy-first AI to partners, clients, and regulators

Regulators increasingly support MPC as an advanced PET. The European Data Protection Board specifically recognizes MPC (“split processing”) as a valid technique for international transfers and compliance-sensitive AI workflows. By keeping data secret, MPC is aligned with key regulatory principles and materially lowers operational risk. Its use signals strong data stewardship and can ease audits and cross-jurisdictional collaboration

Differential Privacy

DP in the AI Lifecycle – DP in Data Collection and Preprocessing

- Noise added before data leaves device or data controller
- Enables privacy-preserving data aggregation from multiple sources
- Supports user-level privacy controls in mobile and IoT applications
- Minimizes risk in sharing and pooling sensitive datasets

By applying differential privacy at data collection points, organizations protect individuals' raw data even before central processing. This is used in mobile devices (e.g., location or health data anonymization), IoT, and multi-party data sharing scenarios. DP-enabled aggregation can combine insights without exposing individual contributions, ensuring privacy by design in data sourcing

DP in the AI Lifecycle – DP in Model Training

- Differentially private algorithms (e.g., DP-SGD) reduce memorization of individual samples
- Noise added to gradients or model updates during training
- Balances trade-off between model accuracy and privacy protection
- Defends against membership inference and model inversion attacks

During model training, DP algorithms add noise in iterative steps (notably DP-stochastic gradient descent) to limit the influence any single data point has on the learned model. This prevents adversaries from reversing engineered models to infer sensitive training data. Proper calibration of DP parameters (epsilon) is essential for maintaining useful model performance while ensuring privacy

DP in the AI Lifecycle – DP in Model Validation and Testing

- Privacy-preserving evaluation of model performance metrics
- Noise ensures no individual test point is identifiable from results
- Enables third-party or external auditing of AI models
- Facilitates use of sensitive validation datasets without exposure

DP is increasingly adopted in model validation to safely assess accuracy, fairness, or robustness metrics without risking revealing individual test or validation data points. This capability supports independent audits or regulators validating compliance without accessing raw data, reinforcing transparency and trust

DP in the AI Lifecycle – DP in Model Deployment and Inference

- Noise injection into inference outputs to protect user queries
- Limits leakage of sensitive input information through model outputs
- Gives users assurance of data confidentiality in AI interactions
- Supports privacy guarantees in services like recommendation systems and digital assistants

In deployment, DP techniques can be applied to add noise in a way that prevents adversaries from extracting private information via outputs, such as recommendations or predictions. This protects users interacting with AI services and helps maintain compliance with privacy laws by ensuring output privacy and limiting overfitting risks

DP in the AI Lifecycle – DP Compliance and Risk Mitigation

- Aligns with GDPR, CCPA, HIPAA data protection mandates
- Serves as key “privacy by design” and “data minimization” tool
- Strong defense against data breach risks and regulatory fines
- Builds user and regulator confidence in AI privacy protections

DP’s mathematically proven privacy guarantees provide legal-safe data handling and AI model design aligned with legislation worldwide. Implementing DP reduces the risk and potential liability from breaches or data misuse. Embracing differential privacy signals commitment to ethical and responsible AI, helping organizations navigate increasingly stringent privacy landscapes

Federated learning

FL in the AI Lifecycle – Initialization and Local Training

- Initialization: Central server distributes initial model and training instructions
- Local Training: Each client trains model using only local data
- Maintains data privacy and autonomy
- Addresses data heterogeneity and scalability challenges

The FL process starts with a global model sent by the central server to clients (devices, institutions). Each client trains the model using its own data, which never leaves local storage. This approach respects data locality and autonomy, essential for privacy compliance and reducing latency. It also effectively handles data that's heterogeneous and distributed across many sources

FL in the AI Lifecycle - FL in Model Aggregation and Iteration

- Clients send local model updates (e.g., gradients) to server
- Server aggregates updates using federated averaging
- Privacy-enhancing techniques like secure aggregation and DP applied
- Updated global model redistributed for next round

Rather than sharing local datasets, clients transmit model updates to the central server. The server aggregates these updates, commonly via weighted averaging, to refine the global model. Privacy-enhancing methods such as secure aggregation ensure individual updates remain confidential. This iterative process continues until performance goals are met, resulting in a robust model trained across distributed data without compromising privacy

FL in the AI Lifecycle - FL Privacy and Security Contributions in AI Lifecycle

- Minimizes risk by avoiding raw data transfer
- Integrates with PETs: DP, Secure MPC, Homomorphic Encryption
- Enables compliance with data protection laws
- Useful in healthcare, finance, IoT, smart city applications

FL significantly reduces privacy risk by design by restricting data movement. It is often combined with other PETs like differential privacy or secure MPC to strengthen guarantees. FL's decentralization supports compliance with regulations like GDPR or HIPAA, making it suited for sensitive domains such as healthcare patient records, financial transactions, or IoT sensor data

FL in the AI Lifecycle - Challenges and Best Practices in AI Lifecycle

- Managing client availability and unreliable connections
- Handling non-IID data distribution and client diversity
- Communication efficiency and system scalability
- Model fairness, explainability, and auditability considerations

Despite advantages, FL faces challenges including intermittent client participation, heterogeneity in local data distributions, and communication overhead. Addressing these requires robust aggregation algorithms, efficient communication protocols, and mechanisms ensuring model fairness and transparency. Best practices include periodic auditing and explainability to boost trust

privacy-preserving synthetic data

CREST

CISA Certified Information Systems Auditor. An ISACA® Certification

bsi ISO/IEC 27001 Information Security Management CERTIFIED

bsi ISO 9001 Quality Management Systems CERTIFIED

Crown Commercial Service

Assentian Limited

Synth Data in the AI Lifecycle - Synthetic Data in Data Collection & Preparation

- Circumvents privacy restrictions on sensitive data use
- Generates large datasets to complement or replace real data
- Enables sharing and pooling across organizations without revealing personal information
- Facilitates safe data distribution in regulated sectors (healthcare, finance)

When real data access is limited by privacy or legal constraints, synthetic data can replicate and extend datasets, allowing AI development to proceed without risking data exposure. It enables collaboration between organizations that cannot share raw data and supports development in regulated environments like health or finance.

Synth Data in the AI Lifecycle - Synthetic Data in Model Training

- Augments original datasets for improved generalization
- Reduces bias and protects against overfitting sensitive examples
- Enables training of AI models without direct use of real personal data
- Facilitates rapid iteration and testing cycles

Using synthetic data alongside or in place of real data improves model robustness by providing a more balanced and extensive training set. It reduces risks of memorizing sensitive information and biases inherent to small datasets. This supports faster experimentation and more privacy-preserving AI model development.

Synth Data in the AI Lifecycle - Synthetic Data in Model Validation and Testing

- Provides diverse, controlled datasets for benchmark testing
- Avoids exposure of private validation data during audits or external reviews
- Allows simulation of rare or extreme scenarios not well represented in real data
- Improves fairness and reliability assessments

Synthetic data is valuable for validating AI models using datasets crafted to test edge cases, bias, and fairness without revealing confidential real data. It enables independent audits and testing while respecting data protection rules, and supports development of trustworthy and safe AI.

Synth Data in the AI Lifecycle - Synthetic Data in Model Deployment and Monitoring

- Simulates input scenarios for ongoing monitoring and anomaly detection
- Supports privacy-preserving model updates and retraining cycles
- Helps generate synthetic transaction or interaction logs for cybersecurity
- Enhances continuous AI system performance while maintaining compliance

Synthetic data can generate simulated workloads and tests to monitor AI behavior in production without risking live data exposure. It can also be used to retrain models or detect unusual activity in AI deployments, enabling privacy-conscious continuous improvement and compliance adherence.

Synth Data in the AI Lifecycle - Synthetic Data Compliance and Risk Mitigation

- Satisfies “data minimization” and “privacy by design” principles
- Helps meet GDPR, CCPA, HIPAA requirements by avoiding real PII use
- Reduces risks of data breaches, leaks, and regulatory penalties
- Improves public trust through ethical data handling practices

By decoupling AI development from real personal data, synthetic data supports legal compliance with international privacy laws and regulations. Its use reduces the attack surface for breaches and data misuse. Organizations using synthetic data demonstrate commitment to privacy, which can bolster reputational and regulatory standing.

Summary - The Overarching Role of PETs in the AI Lifecycle

- Enable privacy-by-design from data sourcing to deployment
- Facilitate secure, compliant sharing and collaboration
- Support bias reduction through synthetic and anonymized data
- Enhance trust and transparency in AI systems

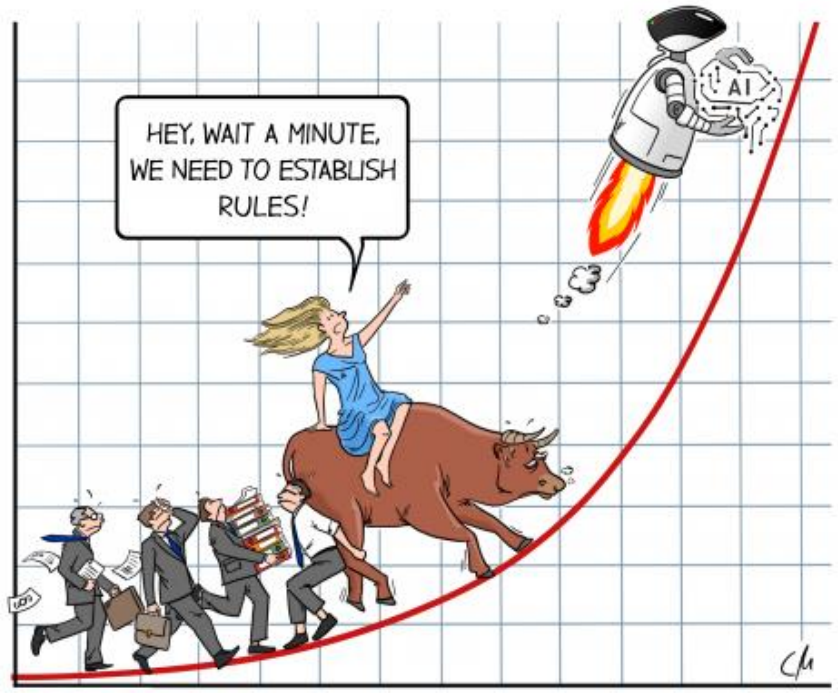
Privacy Enhancing Technologies underpin responsible AI by embedding privacy into every phase of the AI lifecycle—starting with sourcing data, training models, validating, and deploying AI. They enable organizations to share insights securely, reduce bias through synthetic data, and strengthen user trust by providing robust privacy guarantees. PETs make compliance easier and foster innovation while safeguarding individual rights.

Summary - The Multi-Faceted Value of PETs in AI

- Protect sensitive data at every stage: collection, training, validation, deployment
- Enable collaborative AI without risking data leakage
- Close the gap between regulatory compliance and operational performance
- Drive trustworthy AI development with transparency and control

PETs serve as vital tools that secure sensitive data during its entire journey in AI workflows. They unlock collaborative opportunities that were previously impossible due to privacy concerns. By integrating PETs, organizations can meet strict regulations like GDPR or HIPAA while sustaining high-performance AI systems. Ultimately, PETs are essential for building trustworthy AI that aligns with societal, legal, and ethical standards.

Regulatory Aspects



The current state of AI in the compliance landscape

Artificial Intelligence has rapidly moved from theoretical to transformational, profoundly changing how businesses operate across industries. While the benefits of AI, particularly Generative AI (genAI), are monumental, these technologies introduce a new range of risks. Compliance professionals are contending with an evolving regulatory landscape, where staying ahead requires a proactive and informed approach.

The EU Artificial Intelligence Act (European AI Act)

U.S. AI legislation and principles

National Artificial Intelligence Initiative Act of 2020

AI in Government Act & Advancing American AI Act

Blueprint for an AI Bill of Rights Principles

State-Level Legislation

International standards regulating AI

CREST

CISA Certified Information Systems Auditor. An ISACA® Certification

bsi ISO/IEC 27001 Information Security Management CERTIFIED

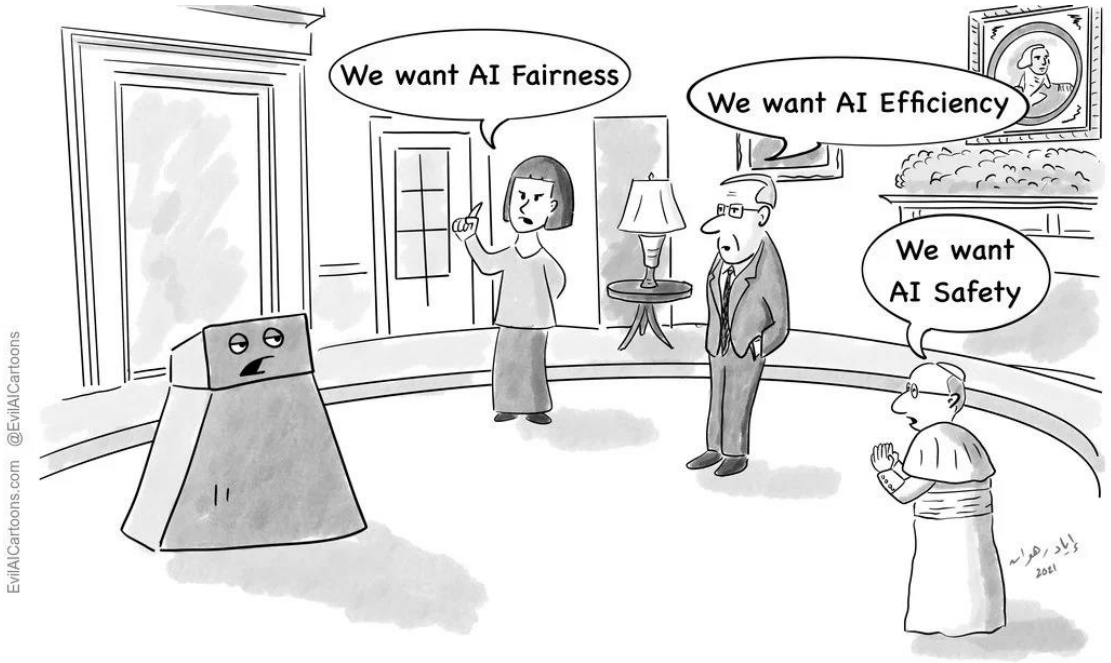
bsi ISO 9001 Quality Management Systems CERTIFIED

Crown Commercial Service

Assentian Limited

Overarching Goals of Responsible AI Regulation

- Ensure AI safety and reliability
- Protect fundamental rights and privacy
- Promote transparency and accountability
- Limit harmful or high-risk AI uses
- Foster innovation while preventing misuse



<< And I want infinite battery! Talk to me when you've negotiated the tradeoffs! >>

Mechanisms to Achieve Regulatory Objectives

- Risk-based classification frameworks for AI systems
- Mandatory transparency, documentation, and explainability
- Requirements for robust governance and human oversight
- Testing, monitoring, and auditing of AI performance
- Compliance with international standards (ISO/IEC 42001, NIST RMF, OECD Principles)

Achieving Public Trust and Market Confidence

- Building citizen confidence in AI outcomes
- Enabling fair market access through unified rulebooks
- Supporting international cooperation & standardization
- Emphasizing ethical, sustainable AI development
- Turning compliance into a ‘trust advantage’

European Union (EU) – AI Act & GDPR

- AI Act (2024): Risk-tiered regulation for trusted and responsible AI
- GDPR: Mandates privacy by design/default, requires PETs for compliant processing
  - PETs role: Differential privacy, federated learning, and homomorphic encryption promoted for high-risk use cases
  - Official guidance: Integrate PETs for anonymization, secure sharing, and robust data governance

The EU AI Act and GDPR jointly set the global benchmark for responsible, privacy-protecting AI. High-risk systems must use techniques like federated learning, differential privacy, and synthetic data for compliance. PETs are cited directly in regulatory guidance especially for functions like data minimization, privacy-preserving collaboration, and data sharing across EU data spaces. This formal integration establishes PETs as practical compliance tools

USA – State Laws & Federal Guidance

- Fragmented landscape: federal executive orders, sector-specific law, multiple new state data privacy statutes
- PETs not mandated but strongly recommended for compliance (CCPA, HIPAA, emerging state laws)
  - Use cases: PETs adopted for sensitive analytics in healthcare, financial services, and cross-border data transfers
  - Trend: Increasing reliance on encryption, confidential computing, federated learning to avoid liability

The US AI regulatory environment is patchwork, with federal recommendations and increasingly strict state laws. While PETs (differential privacy, confidential computing, federated analytics) are not obligatory, best practice guidance from HIPAA, FTC, and state privacy acts (CA, CO, NY) encourage their use for handling protected data. Compelling PET adoption trends arise from regulatory risk and rising costs of breaches

Hong Kong – Privacy Management & Cross-Border Data

- Personal Data (Privacy) Ordinance (PDPO) and recent enhancement bills
- Strong emphasis on privacy management programs and cross-border controls
- PETs valued for data minimization and secure transfer in regional finance, e-commerce
- Regulator (PCPD): Advocates for anonymisation, privacy-preserving analytics, and encrypted data flows

Hong Kong's regulatory model is centered on robust privacy governance, especially for international data transactions. PETs—including data masking, secure multi-party computation, and encrypted data transfer are recommended by regulators for financial systems, telecoms, and e-commerce platforms. PET adoption is framed as an efficiency and compliance accelerator

China – Generative AI and Data Security Law

- Interim Measures for Generative AI Services (2023) + Data Security Law
- PETs: Mandatory data minimization, lawful encrypted processing, secured consent/traceability
- Content providers must utilize PETs for secure ML model training and cross-provincial data sharing
- Emphasis: Homomorphic encryption, differential privacy, and deep anonymisation for national security

China's regulations heavily stress data sovereignty, lawful processing, and national security. Organizations deploying AI—especially generative and cloud services—are required to encrypt, anonymize, and securely manage data, with PETs occupying a central role in regulatory technical guidance for protected machine learning, cross-border flows, and secure cloud collaboration

Singapore – PDPA, AI Governance Model, Sandbox

- PDPA (Personal Data Protection Act) + AI Governance Framework
- Strong focus on privacy by design, sectoral risk assessments
- PETs: MOH/IMDA/PDPC promote use of federated learning, synthetic data, and SMPC for healthcare, finance, smart city analytics
- Regulatory sandbox: Active pilots integrating advanced PETs for compliant real-world deployments

Singapore's comprehensive AI governance model places PETs at the heart of compliant innovation. Sectoral guidelines recommend federated learning, synthetic data, and secure multi-party computation particularly for sensitive smart city, health, and finance AI deployments. Regulatory sandboxes allow real-world PET trials as part of privacy and risk management "by design."

Japan – Sectoral Model, Privacy Guidelines

- Act on the Protection of Personal Information (APPI) amended for AI
- Sector-driven codes: Health, finance, advanced manufacturing
- PETs: Differential privacy and federated analytics prioritized for patient data and industrial IP
- Guidelines: Transparent use of privacy-preserving computation, mandatory security audits

Japan's approach is sectoral, with regulators urging PET use—especially differential privacy and federated analytics—for AI systems processing sensitive personal, medical, or business data. AI guidelines stress the value of secure, privacy-preserving technology adoption and require comprehensive audits of privacy risk, algorithmic bias, and data leakage

Discussion

- **Each Participant**
  - Identify a use case either within the AI Lifecycle or a direct privacy challenge within your organization
  - Identify the specific regulatory requirements as they apply to that use case
  - Define an approach using PETs that takes into account: privacy requirements, data utility requirements, data types, data volumes, centralized or distributed etc etc

- Presentation of Results and Group Discussion

EXPERIENCES WITH PRIVACY ENHANCING TECHNOLOGIES

EUROPEAN CENTRAL BANK | EUROSYSTEM

Digital euro and privacy

The digital euro: privacy by design

To protect your data, we are designing the digital euro to offer the highest privacy levels of any electronic payment option.

Ensuring user privacy has been a central focus of the digital euro project from the start. It requires technological innovation and rigorous compliance with a strong legal framework.

The digital euro would have stringent privacy and inclusion standards, safeguarding user data and rights in the digital age.

Trustworthy AI Center of Excellence

JPMorganChase

Impact of Synthetic Data in Post-Training

Federated learning

DataVaults

TRAMS-AI

EMPOWERING TRUSTED, RESPONSIBLE AI INNOVATION

CREST

CISA Certified Information Systems Auditor. An ISACA® Certification

bsi ISO/IEC 27001 Information Security Management CERTIFIED

bsi ISO 9001 Quality Management Systems CERTIFIED

Crown Commercial Service

Assentian Limited

Science and Technology

DataVaults

Large Synthetic Model capabilities to generate synthetic data that is statistically accurate when real data is unavailable or low in volume, while enhancing data auditing using differential privacy. The solution supports tabular, sequential, relational and text data modalities.

Federated learning

vodafone

CREST CISA Certified Information Systems Auditor. An ISACA® Certification bsi ISO/IEC 27001 Information Security Management CERTIFIED bsi ISO 9001 Quality Management Systems CERTIFIED Crown Commercial Service

Assentian Limited

DataVaults

Andaman7

Westpac

Telstra

comune di PRATO

ATHENS INTERNATIONAL AIRPORT ELEFTHERIOS VENIZELOS

bamboo energy

CREST

CISA Certified Information Systems Auditor. An ISACA® Certification

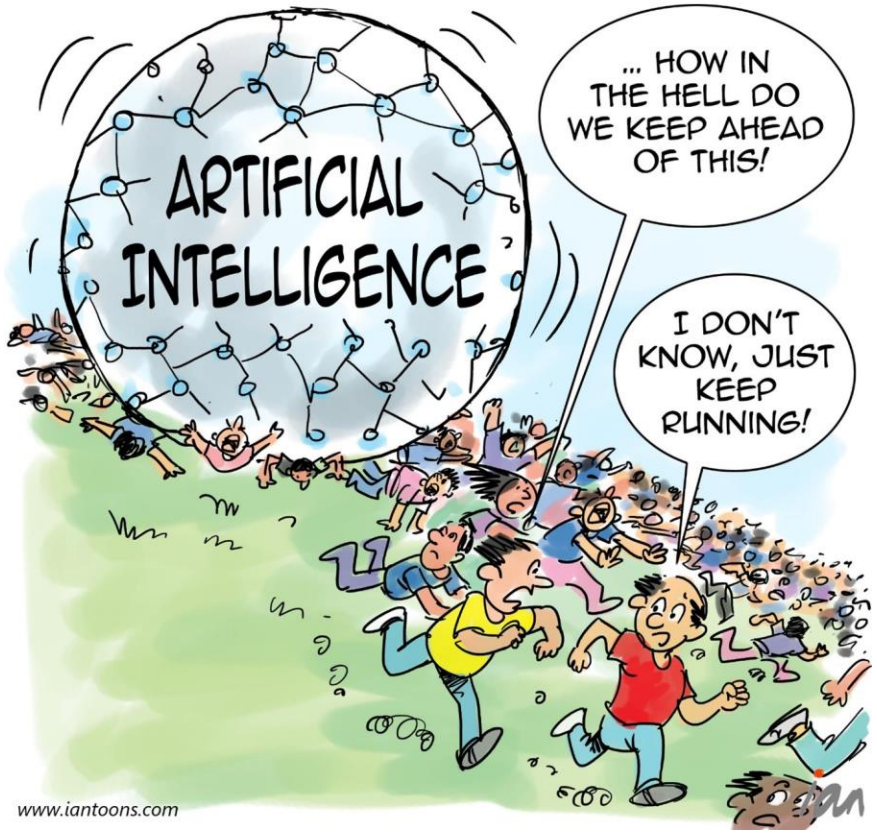
bsi ISO/IEC 27001 Information Security Management CERTIFIED

bsi ISO 9001 Quality Management Systems CERTIFIED

Crown Commercial Service

Assentian Limited

Future Trends



SHARING TRUSTWORTHY AI MODELS WITH PRIVACY-ENHANCING TECHNOLOGIES

OECD ARTIFICIAL INTELLIGENCE PAPERS
June 2025 **No. 38**

Use case archetype 1 — Enhancing AI model performance through minimal and confidential use of input and test data: Often, no single organisation will have access to the necessary variety or volume of data, making external data access critical yet complex due to confidentiality, trust, or regulatory concerns. In the use cases highlighted during the OECD expert workshops on PETs

Use case archetype 2 — Enabling confidential co-creation and sharing of AI models: In addition to the collaborative use of input and test data, the joint development of AI models by multiple parties as well as their joint re-use (AI model sharing) can drive innovation by e.g. democratising access to powerful AI models, including by smaller actors such as small and medium-sized enterprises (SMEs). However, co-creating or sharing AI models presents heightened confidentiality risks, as shared models can be the target of unauthorised access, manipulation, or extraction of confidential proprietary or personal information (through e.g. reverse engineering from model weights) with negative effects on privacy and other rights and interests. Under this archetype, PETs are primarily used to: (i) confidentially co-create AI models, with PETs enabling distributed confidential data processing, such as MPC and federated learning, being the most prominent PETs; and (ii) protect AI models and their outputs (inferences) by providing complementary layers of protection. For example, differential privacy helps reduce the identifiability of output data, while trusted execution environments (TEEs) and (fully) homomorphic encryption (HE) protect the confidentiality of data and models during computation.

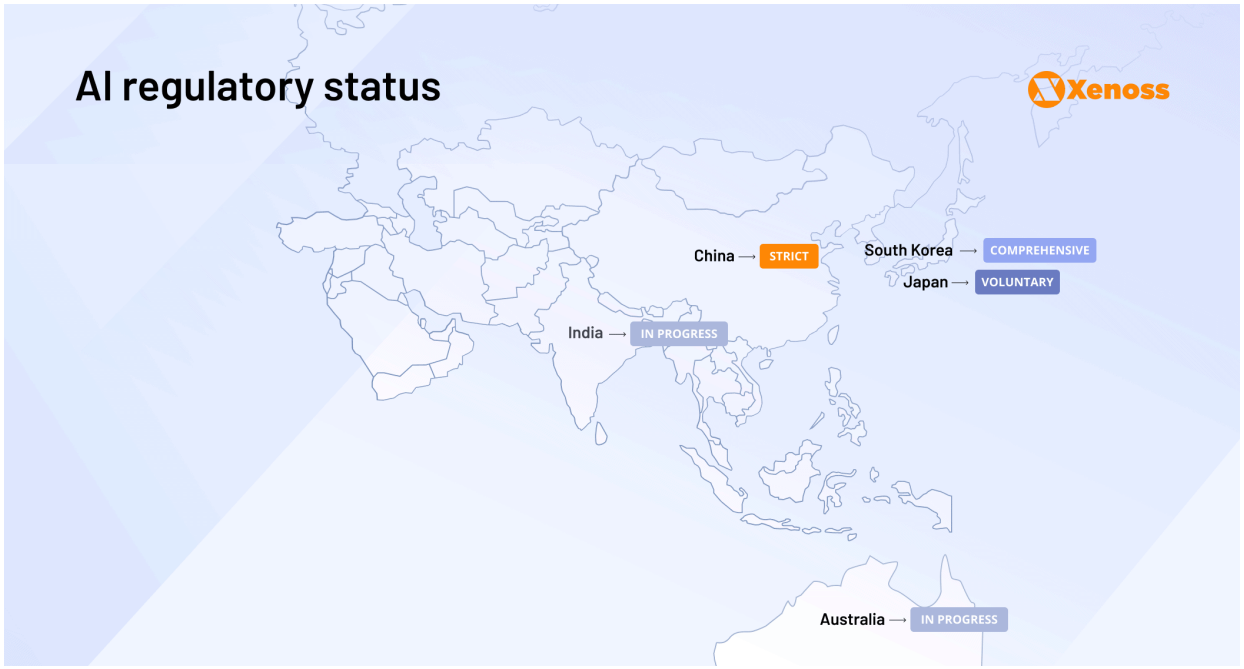
Top 5 Data Privacy trends in 2025

TrustCloud

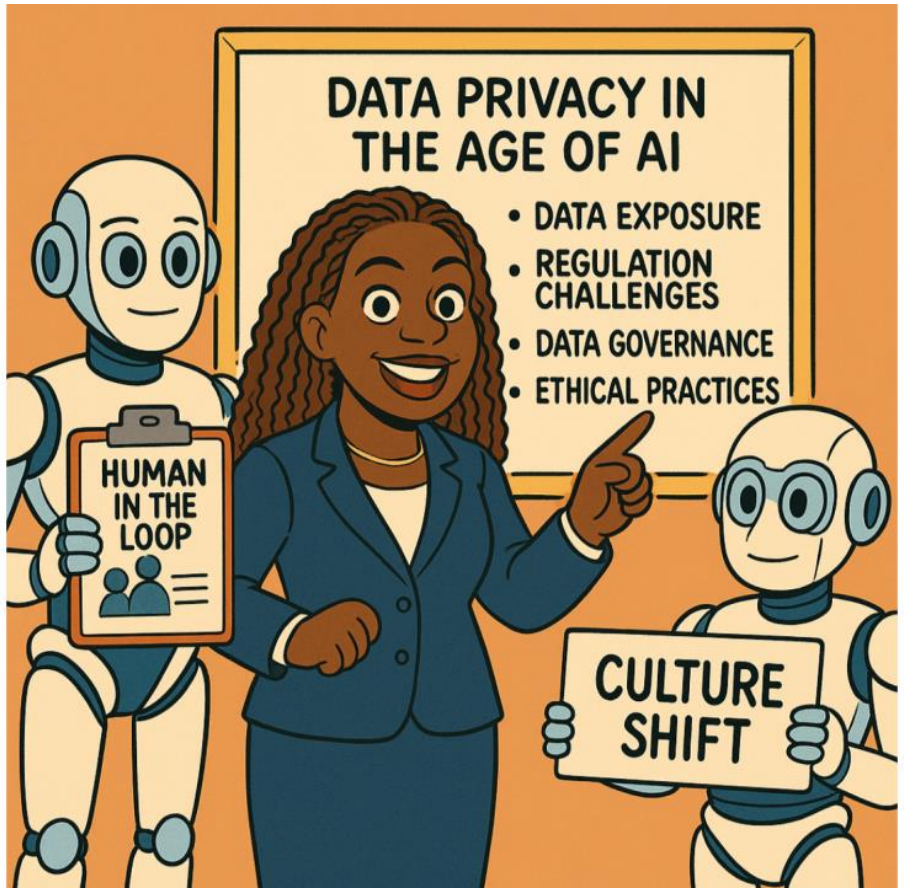
Here are some data privacy trends that might be relevant and might continue to shape the landscape in 2025. These trends may have evolved or changed.
Following are the top 5 Data Privacy trends in 2025:

1. Increased data protection regulations and compliance
2. Focus on data breach prevention and incident response
3. Enhanced consent management and user control
4. Rise of privacy enhancing technologies (PETs)
5. Focus on ethical data use and AI

www.trustcloud.ai



In September 2024, the Australian government released a **Voluntary AI Safety Standard** comprising a number of AI guardrails to create best practice guidance for the use of AI. The government also proposed **mandatory guardrails for AI in high-risk settings**, which were subject to public consultation. It is possible Australia could enact legislation drawing upon some of the concepts in the EU AI Act, but it currently remains unclear how the government will proceed. In May 2024, the Singapore government introduced the **Model AI Governance Framework for Generative AI**, which details best practice guidance on responsible development, deployment and use of AI. China's **Interim Measures for the Management of Generative AI Services** commenced in 2023 and should continue to be observed as the region's first comprehensive binding regulation on generative AI.


DATA PRIVACY IN THE AGE OF AI
• DATA EXPOSURE
• REGULATION CHALLENGES
• DATA GOVERNANCE
• ETHICAL PRACTICES
HUMAN IN THE LOOP
CULTURE SHIFT

How Generative AI is Changing Data Privacy Expectations

Changing privacy expectations

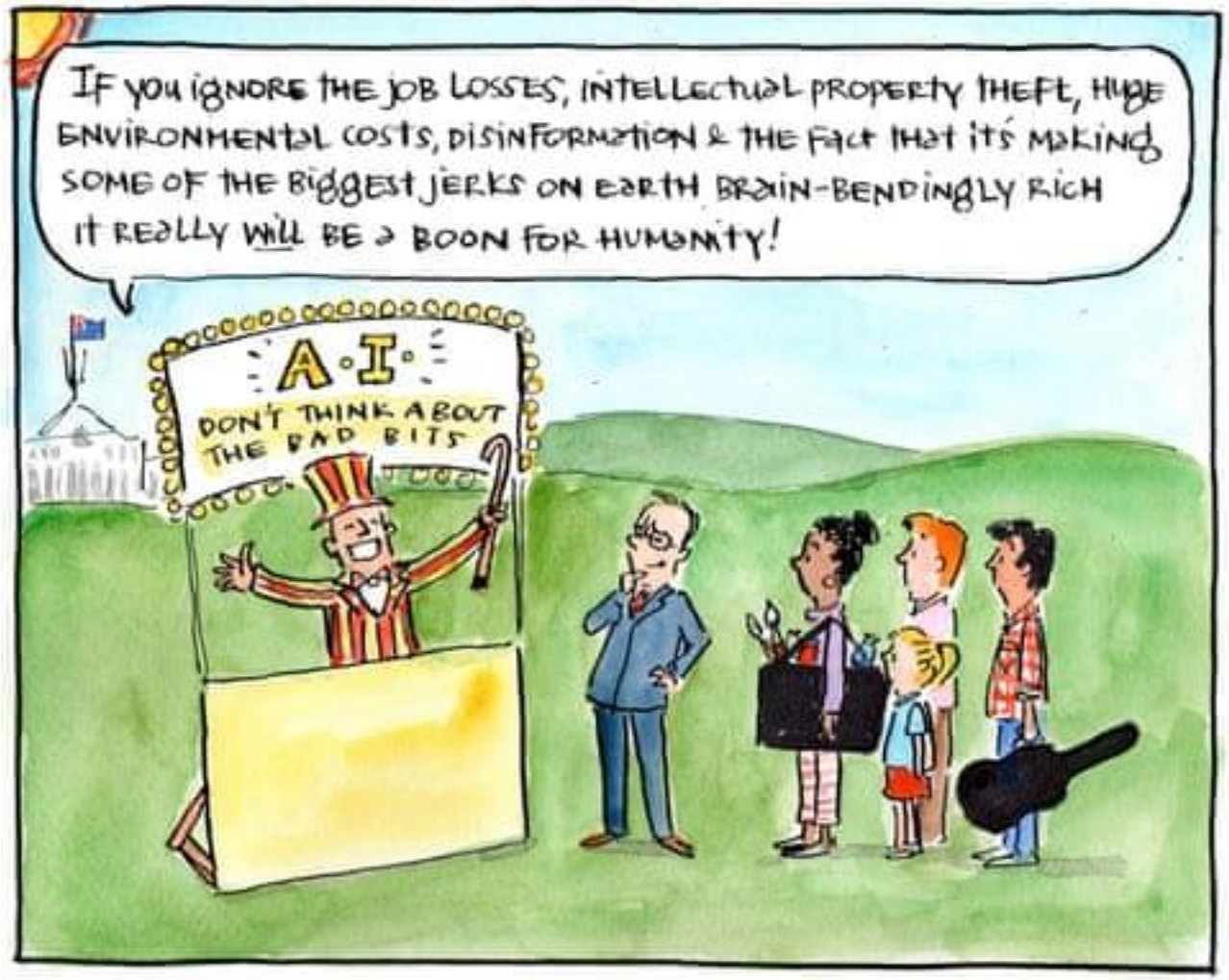
Generative AI systems are trained on large-scale datasets that may include everything from public internet content to user-generated data. This creates a seismic shift in privacy expectations:

- **Unintentional exposure:** AI models might inadvertently regurgitate personal data on which they were trained.
- **Purpose drift:** Data collected for one reason might be used in entirely different ways through generative models.
- **Perpetual processing:** AI systems often retain information in ways that make it difficult to trace or erase.

Privacy as a priority

To mitigate risk and maintain trust, organizations must treat data privacy not as an afterthought but as a foundational pillar in AI adoption. This includes integrating privacy by design principles, conducting AI-specific privacy impact assessments, and ensuring transparency in how AI systems use data.

Supporting this shift, the 2024 TrustArc Global Privacy Benchmarks Report found that **AI remains the top privacy challenge for organizations worldwide for the second consecutive year.** Additionally, **70% of companies identified AI as an important or very important privacy concern**, underscoring how AI-related risks are shaping strategic data privacy priorities.



Group Exercise: Exploring Future Trends in AI and the Role of PETs

Collaborative exploration of emerging AI trends and the evolving role PETs will play in enabling responsible, trusted AI.

Form Groups: Divide participants into groups of 4-6 based on equal distribution.

Assign Topics: Assign or let each group choose one of the following future AI trends to discuss:

- Multi-modal and embodied AI systems
- Agentic and autonomous AI systems
- Federated learning and decentralized AI ecosystems
- Integration of AI with blockchain and digital identity
- Advances in PETs like post-quantum cryptography and composable privacy
- AI governance, auditability, and regulatory evolution

Discussion Points: Each group should discuss the following for their trend:

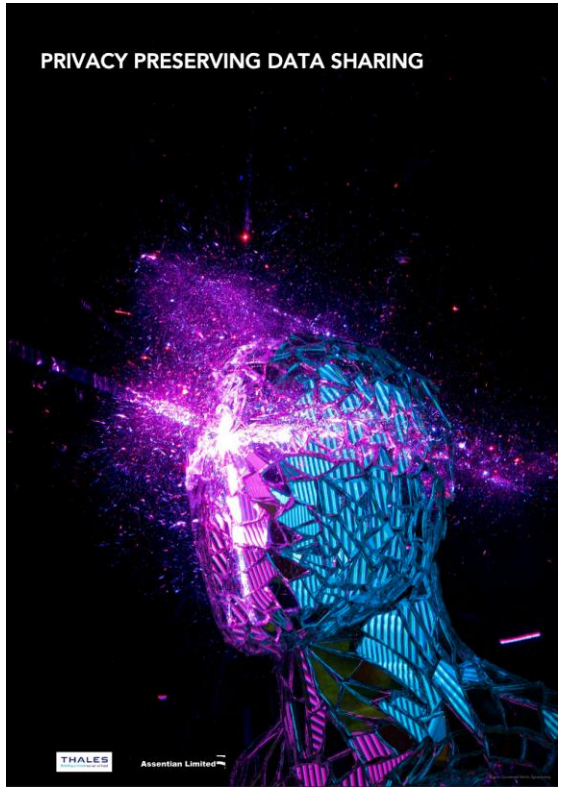
- What are the key characteristics and potential impacts of this trend?
- How critical is the role of PETs in enabling or securing this trend?
- What PETs are most relevant or likely to evolve for this trend?
- What are major challenges and risks related to privacy, security, and adoption?
- How should organizations prepare and adapt to these trends with PETs?

Group Presentations: Each group presents their findings to the whole workshop. Encourage Q&A and cross-group discussion.

Summary Video:
How PETs make AI Trustworthy

CREST
CISA Certified Information Systems Auditor. An ISACA® Certification
bsi ISO/IEC 27001 Information Security Management CERTIFIED
bsi ISO 9001 Quality Management Systems CERTIFIED
Crown Commercial Service

Assentian Limited



CONTACT

CONTACT US

https://www.assentian.com/

DataVaults

TRAMS-AI

EMPOWERING TRUSTED, RESPONSIBLE AI INNOVATION

CREST

CISA Certified Information Systems Auditor. An ISACA® Certification

bsi ISO/IEC 27001 Information Security Management CERTIFIED

bsi ISO 9001 Quality Management Systems CERTIFIED

Crown Commercial Service

Assentian Limited