Pre-standardization report on:

Study of De-identification of training data for ML

Under the guidance of

Kshitij Bathla

Scientist-C/Deputy Director

Electronics & IT Department

Submitted By:

Debolina Ghosh

Intern, BIS

Acknowledgement

I would like to extend my heartfelt gratitude to the Bureau of Indian Standards (BIS) for granting me the opportunity to undertake this internship and contribute to the Study of de-identification of training data for ML. Special thanks to Shri Kshitij Bathla, Scientist-C and Deputy Director, Electronics & IT Department, for his invaluable mentorship, guidance, and unwavering support throughout the entire duration of this internship. I am deeply appreciative of Shri Kshitij Bathla's expertise and dedication, which have been instrumental in shaping the scope and direction of this study. His insightful feedback and constructive critiques have immensely enriched the quality and depth of the research.

I also want to express my sincere appreciation to the Head of Electronics and IT Department, Mrs. Reena Garg, Scientist G at BIS, for creating a conducive and collaborative environment. The opportunities for knowledge-sharing and exposure to real-world challenges have been truly rewarding.

Furthermore, I am indebted to all the experts and stakeholders in the field who generously shared their insights and expertise during the visit at National Informatics Centre (NIC). Their inputs have played a pivotal role in shaping the methodology and conclusions of this report.

Lastly, I wish to acknowledge the support and encouragement of my family and friends. Their belief in my abilities and constant motivation have been a driving force throughout this internship.

Thank you.

Sincerely,
Debolina Ghosh
Email: dghosh_be21@thapar.edu
Intern, Bureau of Indian Standards (BIS)

Table of Contents

1.Introduction

The explosive growth of machine learning has made it a critical infrastructure in the era of artificial intelligence. The extensive use of data poses a significant threat to individual privacy, prompting various countries to implement corresponding laws, such as GDPR, to protect individuals' data privacy and the right to be forgotten. The rise of machine learning has been nothing short of revolutionary, transforming the way we extract insights and understand the world around us. With its ability to uncover patterns and automate complex tasks, this powerful technology has become an indispensable tool in our data-driven era.

Yet, as we marvel at the incredible potential of machine learning, we find ourselves confronted with a critical challenge: how can we harness this power while ensuring the privacy and security of the sensitive information that fuels it? This is where privacy-preserving machine learning comes into play. It's a field that gained considerable traction in recent years, as researchers and practitioners strive to develop techniques that allow us to harness the power of machine learning while safeguarding the confidentiality of our data.

We live in an era where data breaches and unauthorized access to sensitive information are all too common. In addition to the threats of illegitimate access to data through security breaches, machine learning models pose an additional privacy risk to the data by indirectly revealing about it through the model predictions and parameters.[1] This not only poses a risk to individuals, but can also have significant implications for businesses, governments, and other organizations that rely on data-driven decision-making. Moreover, as machine learning models become more sophisticated and are used in an ever-expanding range of applications, the potential for misuse or mishandling of sensitive data grows. This is particularly true in fields like healthcare, finance, and national security, where the stakes are high and the consequences of data breaches can be severe. [2]

Machine learning algorithms can leak a significant amount of information about their training data. A legitimate user of a model can reconstruct sensitive information about the training data by having access to its predictions or parameters. Due to the important role of machine learning algorithms, abundant computer systems are beginning to hold a large amount of personal data for decision-making and management. For example, the increasingly notable ChatGPT actively utilizes large datasets for knowledge discovery. However, research has shown that machine learning models can remember information about training data, raising concerns about potential attacks on individual privacy.

Adversarial attacks, such as membership inference attacks and model inversion, have demonstrated the ability to extract information about target data from machine learning models. In response to these concerns, there have been significant developments in regulations and laws governing individual privacy. For instance, the General Data Protection Regulation (GDPR) implemented by the European Union, and the California Consumer Privacy Act (CCPA) specifically state the right to be forgotten. Data owners are then obligated to respond to these deletion requests promptly, leading to the need for privacy methods.

Our search queries, browsing history, purchase transactions, the videos we watch, and our movies' preferences are but a few types of information that are being collected and stored on a daily basis. This data collection happens within our mobile devices and computers, on the streets, and even in our own offices and homes. Such private data is being used for a variety of machine learning applications. Machine learning (ML) is being increasingly utilized for a variety of applications from intrusion detection to recommending new movies. Some ML applications require private individuals' data. Such private data is uploaded to centralized locations in clear text for ML algorithms to extract patterns and

build models from them. The problem is not limited to the threats associated with having all this private data exposed to insider threat at these companies, or outsider threat if the companies holding these datasets were hacked. [3]

Machine learning (ML) is increasingly being adopted in a wide variety of application domains. Usually, a well-performing ML model, especially emerging deep neural network models, relies on a large volume of training data and high-powered computational resources. The need for a vast volume of available data raises serious privacy concerns because of the risk of leakage of highly privacy-sensitive information and the evolving regulatory environments that increasingly restrict access to and use of privacy-sensitive data. Furthermore, a trained ML model may also be vulnerable to adversarial attacks such as membership/property inference attacks and model inversion attacks.[4]

De-identification is one of the key techniques used to mitigate these risks. By removing or masking personally identifiable information (PII) from datasets, de-identification helps ensure that the data cannot be traced back to specific individuals. However, de-identification alone may not be sufficient, as sophisticated attacks can sometimes re-identify anonymized data. Therefore, combining de-identification with other privacy-preserving techniques, such as differential privacy and secure multi-party computation, provides a more robust approach to protecting sensitive information.[9]

By understanding these challenges and implementing privacy-preserving techniques, we can continue to benefit from the advances in machine learning while ensuring the privacy and security of sensitive information.

2. Scope

This study explores the integration of privacy-preserving techniques into machine learning processes to address the critical issue of data privacy in the modern digital era. The focus will be on understanding various privacy concerns and threats associated with machine learning and analysing various methods to protect data privacy, with a particular emphasis on de-identification techniques, which are among the most commonly used approaches for ensuring privacy of training data machine learning.

3. Identification of sensitive data in Machine Learning

In certain instances, identifiability of the PII principal might be very clear (e.g., when the information contains or is associated with an identifier which is used to refer to or communicate with the PII principal). Information can be considered to be identifier under listed categories [5]

- Personal identifiable information, such as full names, date of birth, address, and phone number.
- Social data such as ethnicity, religion, sexual orientation, and political affiliation.
- Financial data such as credit card numbers, income, and tax records.
- Health and medical data, such as diagnoses, prescriptions, and genetic data.
- Geolocation data, such as tracking data from mobile devices.
- Biometric data, such as facial recognition, voice, and fingerprints.
- User authentication data, such as usernames, passwords, security questions and answers.
- Legal data, including intellectual property and trade secrets.

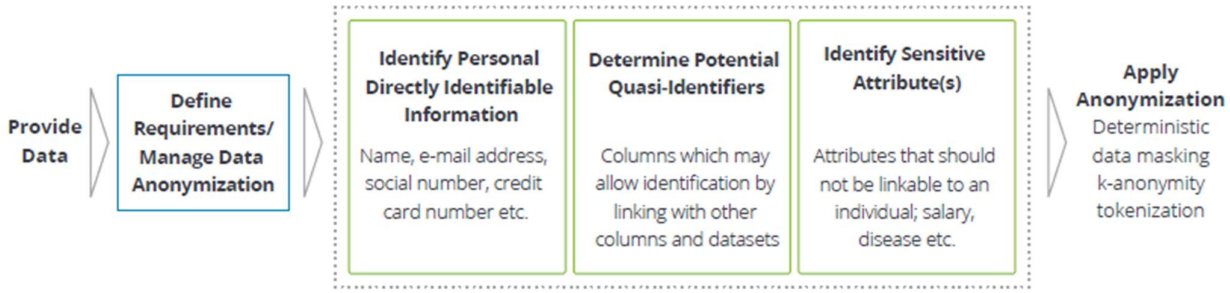

Fig 1: Steps of identification of sensitive data in ML

4. Privacy Concerns relating training data:

When developing ML models for AI systems, there is always a concern that models can leak sensitive information about individual users training data instances. Attackers can find out information about specific users by a different linkage attack [6].

4.1 Membership Inference attack

In machine learning, a membership inference attack (MIA) is a privacy attack that infers the victim model and extracts its training data, privacy settings, and model parameters. In this type of attack, the adversary has access to query the victim model under attack and can analyse the output gathered from the queried results. The adversary can regenerate the training dataset of the targeted adversarial machine learning model by analysing the gathered queried results. It allows an attacker to determine whether a specific data point (record) was part of the training data used to build the model.[7] The attack surface for membership inference attacks on machine learning is highlighted in Fig 2. For instance, in a healthcare scenario, an adversary might infer whether a specific dental X-ray was used to train a model, thereby revealing that the individual is a patient at the hospital. Outliers in the training data are often more susceptible to these attacks, making it crucial to address this vulnerability to protect individual privacy.[6]

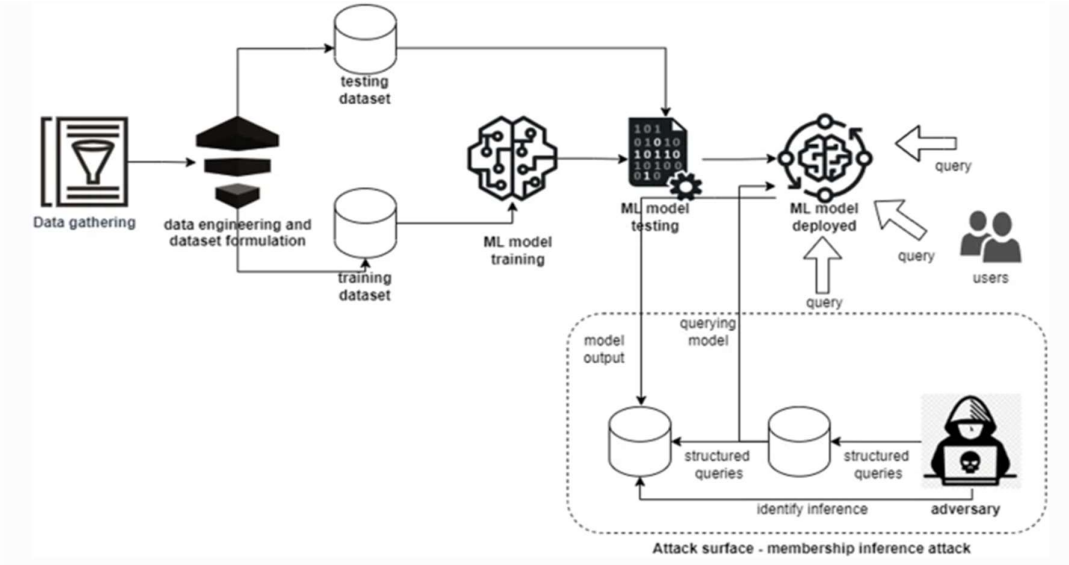

Fig 2: Membership Inference Attack

4.2 Training data extraction

These attacks aim to recover specific examples from the training dataset of a model, potentially exposing sensitive information memorized during training. For example, large language models trained on extensive text data can inadvertently memorize and expose personal identifiers, such as social security numbers. An attacker can extract this information by querying the model and analyzing its outputs, posing a significant risk when such models are used in applications like chatbots that interact with users.[6]

4.3 Embedding inversion attack

Embedding models, which are commonly used in AI systems to handle large inputs efficiently, face multiple attack vectors. Embedding inversion attacks attempt to reconstruct input data from embedding vectors, potentially revealing sensitive information like facial features from image embeddings. Sensitive attribute inference attacks aim to infer private attributes about the input data from the embeddings. Additionally, these models are also susceptible to membership inference attacks, where the goal is to determine if a specific data instance was part of the training set, leading to potential privacy violations.[6]

4.4 Attribute inference attacks

In attribute inference attacks (also called “feature reconstruction attack), the adversary knows some attributes about given data and by attacking a model that was trained on these data, the attacker aims to extract other attributes about those data (*e.g.*, an attacker knows the name and age attributes and aims to infer the gender). This kind of attack targets particularly vertical federated learning settings, where the attacker is either the active party or the passive party [8]. These attacks pose a privacy risk by revealing information about the dataset's composition, which could be used to deduce additional sensitive details or highlight biases in the model [6].

4.5 Model inversion attack

The objective of this attack is to disrupt the privacy of machine learning. Model inversion attack is the type of attack in which an adversary tries to steal the developed ML model by replicating its underlying behaviour, querying it with different datasets. An adversary extracts the baseline model representation through a model inversion attack and can regenerate the training data of the model.[7] For instance, given a model whose output is a user embedding, a model inversion attack might invert the embeddings to reconstruct the user profiles used to train the model. These attacks highlight the risk of revealing sensitive information through the model's responses, necessitating techniques to mitigate such vulnerabilities and protect individual privacy.[6][15]

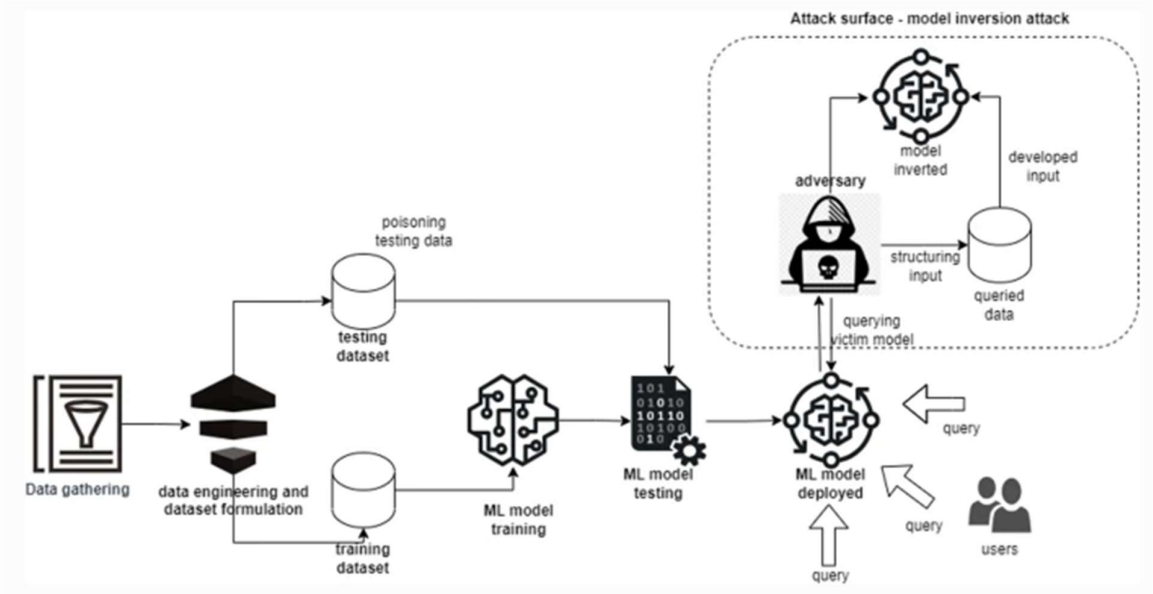

Fig 3: Model Inversion Attack

Privacy attacks		Machine learning		
		Goals	Targets	Workflow
Information -stealing	Membership inference attack	Extracts membership information of training data	Training data	Constructs attack model using outputs of the trained model
	Attribute inference attack	Extracts attribute information of training data	Training data	
	Model inversion attack	Reconstructs the training data	Training data	
	Model extraction attack	Copies the trained model	Trained model	

Fig 4: Data privacy attacks in Machine Learning

5. Privacy-Preserving Techniques in Machine Learning

5.1 Mitigation techniques during the data preparation phase

The data preprocessing or preparation phase is a crucial initial step in the machine learning pipeline, where data are cleaned, labelled if needed, and made ready for model training. In this phase, adopting privacy-preserving technologies aims to minimize data exposure to potential threats by either concealing sensitive attributes or adding noise before sending data to the training computation party. Privacy-preserving approaches for data preparation focus on three main directions:(i) Identifying sensitive attributes and concealing them partially or fully; (ii) Applying perturbation techniques that add a certain amount of noise to prevent reverse engineering statistical conclusions to retrieve original data points; (iii) Resorting to surrogate dataset techniques.[8]

5.1.1 De-identification techniques

"De-identification" is the process of removing the association between a set of identifying data and the data subject. The fundamental objective of de-identification is to protect the privacy of individuals because once de-identified, a dataset is considered to no longer contain personally identifiable information (PII). If a dataset does not contain PII, its use or disclosure will not violate the privacy of individuals. When using data including personal information for research, marketing, testing applications, statistical trending or other legitimate purposes, the involvement of specific individuals has to be clarified in order to meet the data usage goals. In such cases, de-identification of PII is highly recommended.[14]

5.1.1.1 Redaction

It involves the removal or obscuring of sensitive information from a document or dataset. This process ensures that any personally identifiable information (PII) is not visible or accessible. Redaction can be performed manually or using automated tools, and it typically involves blacking out or deleting specific sections of text, images, or other data elements that contain sensitive information. The primary goal of redaction is to protect privacy while retaining the document's utility for authorized personnel.[10]

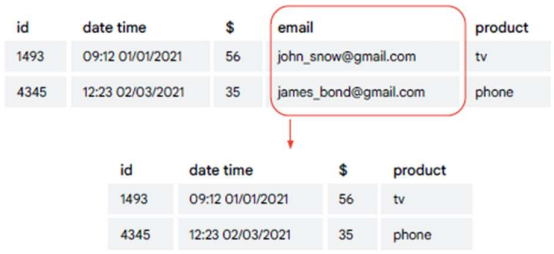

Fig 5: Redaction

5.1.1.2 Replacement

It is the process of substituting sensitive data with non-sensitive values or placeholders. This technique maintains the structure and format of the original data while removing any direct identifiers. For instance, replacing a person's name with a pseudonym or substituting a credit card number with a fictitious number. Replacement can be static or dynamic. Static replacement uses predefined values, while dynamic replacement may generate values based on specific rules or algorithms. This method is particularly useful for testing and training data, allowing datasets to be used without exposing real sensitive information.[10]

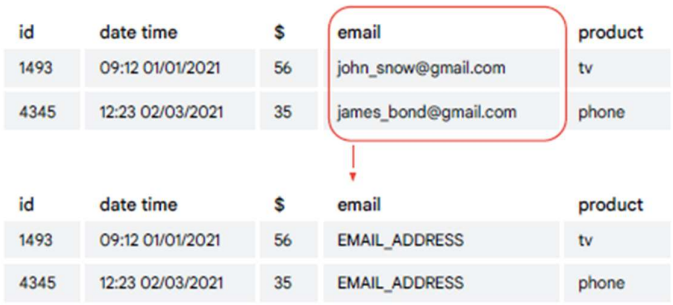

Fig 6: Replacement

5.1.1.3 Data Masking

Involves modifying sensitive data to make it unreadable or obfuscating it so that it cannot be easily understood without a decryption key or algorithm. It is used to protect data in non-production environments, such as development, testing, or training scenarios, where real data is not necessary. Techniques include masking characters, encrypting data, or using algorithms to transform data. Data masking ensures that sensitive information remains secure while allowing the data to be used for its intended purpose without compromising privacy.[10]

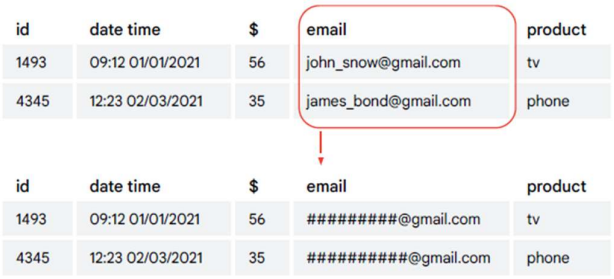

Fig 7: Data Masking

5.1.1.4 Generalisation

Generalisation is the process of removing a value from its accuracy in order to obtain a more general value. It can be applied in ascending order. For example, a full date can be aggregated into a month and a year, which in turn can be aggregated into a year, which in turn can be aggregated into a five-year interval, a ten-year interval and so on. [11]

5.1.1.5 Bucketing

Bucketing involves grouping data values into predefined ranges or buckets. This technique is commonly used for anonymizing numerical data, such as ages, income, or scores, by categorizing them into intervals rather than using specific values. For instance, instead of listing exact ages, data could be grouped into buckets like 20-29, 30-39, etc. Bucketing reduces the granularity of the data, making it harder to identify individual records while preserving the overall data distribution. This method is useful in statistical analysis and data sharing, where maintaining privacy is crucial without significantly affecting the utility of the data.[10]

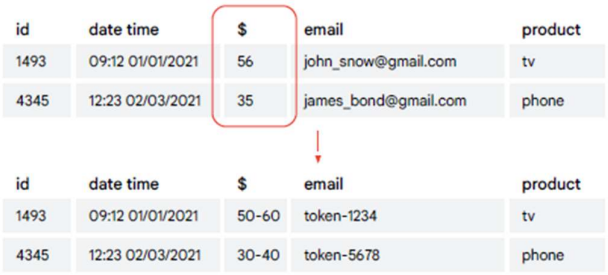

Fig 8: Bucketing

5.1.1.6 Shifting

Shifting involves altering the values of data points by a consistent, random, or calculated amount, such as adding or subtracting a fixed number. This technique is commonly used to obscure exact values while maintaining the relative relationships between data points. For example, shifting ages by a constant number of years or modifying numerical values in a dataset by a certain range. Shifting can be useful for protecting privacy in datasets while ensuring that the data remains usable for analysis, especially in cases where the exact values are not critical but the patterns and relationships need to be preserved. This method is often used in statistical analysis and machine learning to enhance privacy without significantly distorting the data.[10]

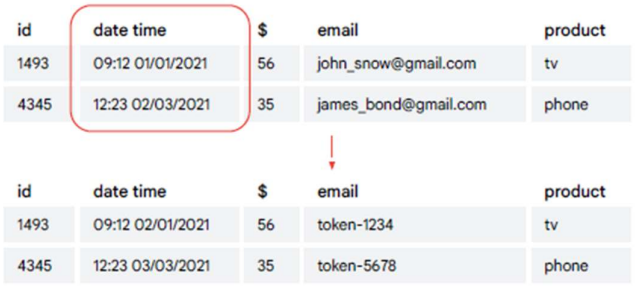

Fig 9: Shifting

5.1.2 Randomization techniques

Anonymization by removing personal identifiers before releasing the data to the public may seem like a natural approach for protecting the privacy of individuals. Indeed, some companies attempted to protect their users' privacy by only releasing anonymized versions of their datasets, as was the case with the anonymous movie ratings released by Netflix to aid contestants for its 1M$ prize to build better recommender systems (for movies). Despite the anonymization, researchers were able to utilize this dataset along with IMDB background knowledge to identify the Netflix records of known users, and were further able to deduce the users' apparent political preferences1. This incident demonstrates that anonymization cannot reliably protect the privacy of individuals in the face of strong adversaries.[43]

5.1.2.1 Data perturbation

This involves altering the original data in a controlled manner to prevent the disclosure of sensitive information. The goal is to balance data utility and privacy by introducing randomness that obscures the true values but maintains the overall statistical properties of the data. This randomness can be introduced through various methods, including:

-Adding Noise: One common method is to add random noise to the data. This noise can be drawn from various probability distributions, such as the Laplace or Gaussian distributions. The amount and type of noise added depend on the desired level of privacy and the sensitivity of the data.

-Data Shuffling: Another approach is to shuffle data entries, which involves randomly permuting the values within a dataset. This technique preserves the overall statistical characteristics of the data but makes it harder to trace specific values to individuals.

-Data Substitution: Involves replacing sensitive data values with synthetic or altered values that are statistically similar to the original values but do not reveal specific information about individuals.

5.1.2.2 Differential Privacy

Differential Privacy is defined by a mathematical framework designed to provide strong guarantees about the privacy of individual data points within a dataset. The core principle of DP is that the output of any computation (or the result of a query) should be nearly indistinguishable whether or not any single individual's data is included in the dataset.[12]

Privacy Parameter ϵ\epsilon: The parameter ϵ\epsilon represents the privacy level, with smaller values indicating stronger privacy. It measures the worst-case ratio of probabilities that an adversary could distinguish between the presence and absence of a particular individual's data. A lower ϵ\epsilon implies that the output is less sensitive to any single record, thus offering greater privacy protection. **Sensitivity**: Sensitivity quantifies how much the output of a function can change when a single individual's data is added or removed. It is crucial in determining the amount of noise to add. Listed are the mechanisms to achieve Differential Privacy:

(i) Adding Noise: The most common technique to achieve DP involves adding noise to the output of a function or query. This noise is typically generated from a probability distribution that depends on the sensitivity of the function. Common distributions used are:

- **Laplace Distribution**: For functions with bounded sensitivity, noise is added from a Laplace distribution with scale parameter proportional to the sensitivity and ϵ(epsilon).
- **Gaussian Distribution**: For more advanced applications or where the privacy parameter ϵ\epsilon is not strict, noise from a Gaussian distribution is used. The variance of the Gaussian noise is proportional to the sensitivity and a function of ϵ\epsilon.

(ii) Differentially Private Stochastic Gradient Descent (DP-SGD): In machine learning, DP can be applied during model training via techniques such as DP-SGD. This involves adding noise to the gradients during the optimization process to ensure that individual training examples do not overly influence the model. Key steps include:

- **Gradient Clipping**: Limiting the norm of gradients to prevent any single data point from having too large an impact.
- **Noise Addition**: Adding noise to the clipped gradients to ensure that the updates remain private.

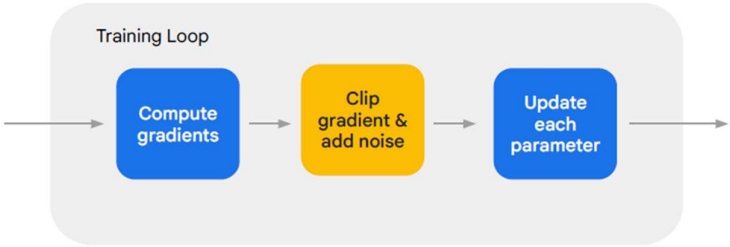

Fig 10: DP-SGD Algorithm add noise to introduce differential privacy

5.2 Mitigation techniques during the Data Training phase

Privacy enhancing techniques added during model training can be divided into four categories: (i) perturbation techniques added to the training algorithm itself to turn it into a differentially private algorithm; (ii) training on encrypted data; (iii) privacy-preserving architectural choices like PATE and federated learning; and (iv) training on privacy-preserving hardware solutions such as trusted execution environments. While the former serve to offer privacy guarantees by mitigating model reverse engineering attacks such as membership inference attacks, the latter offers confidentiality guarantees. This is due to the fact that cryptographic tools, TEEs, and vanilla federated learning guarantee the secrecy of the data and thus prevent attacks that try to directly access the data in its original form, while DP and PATE guarantee that adversaries cannot reveal sensitive information about the data through seeing statistics and/or knowledge extracted from the data.[8]

5.2.1 Federated Learning

Federated Learning is a decentralized approach where multiple participants (devices or servers) collaboratively train a machine learning model while keeping their data local. Instead of sending raw data to a central server, participants send model updates (such as gradients) to the central server, which aggregates these updates to improve the global model. The decentralised training opens many possibilities in real-world applications in which regulations and data sharing policies do not allow the transfer of such sensitive data but only transmitting model parameters.[8] Raw data remains on local devices, reducing the risk of exposure and ensuring compliance with data protection regulations. The learning process is distributed across multiple devices or nodes, which can be heterogeneous in terms of data and computational power. The central server aggregates updates from various participants to update the global model, ensuring that individual data contributions are anonymized [10]. The Federated Learning process typically involves the following steps:

-Initialization: A global model is initialized and distributed to all participating devices.

-Local Training: Each device trains the model locally on its own data, generating model updates (e.g., gradients).

-Update Aggregation: The central server collects and aggregates the updates from all devices. This is usually done by averaging the gradients or model weights.

-Global Model Update: The aggregated updates are used to improve the global model, which is then redistributed to all devices for further local training.

-Iteration: This process is repeated iteratively, with the global model continually improving through the collaborative efforts of all participants.

5.2.2 PATE

Private Aggregation of Teacher Ensembles (PATE) is an innovative framework designed to preserve the privacy of training data in machine learning (ML) systems.The PATE architecture involves training multiple teacher models on disjoint subsets of a private dataset, ensuring that sensitive information is not concentrated in any single model. After training, these teacher models transfer their collective knowledge to a student model by labeling a public, unlabeled dataset, which is then used to train the student. To protect the privacy of the sensitive data the teachers were trained on, differential privacy (DP) noise is added during the labeling process, preventing the student model from inferring sensitive

information. PATE guarantees privacy by limiting the number of teacher votes and revealing only the topmost vote after adding random noise.[8]

5.2.3 Homomorphic Encryption

Homomorphic encryption (HE) is a cryptographic technique that allows computations to be performed on encrypted data without decrypting it. The result of the computation is still encrypted and can be decrypted only by the data owner. This technique is particularly useful in scenarios where sensitive data needs to be processed by untrusted parties, such as in cloud computing environments.[9]

Mechanism:

- **Encryption Scheme**: HE schemes support different types of operations, such as addition (additive homomorphism) and multiplication (multiplicative homomorphism). Fully Homomorphic Encryption (FHE) supports both types of operations, enabling arbitrary computations.
- **Ciphertext Operations**: In HE, operations are performed directly on ciphertexts. For example, given encrypted values E(m1) and E(m2), where m1 and m2 are plaintext values, an operation such as addition on these ciphertexts will result in an encrypted value of m1+m2.
- **Decryption**: After computation, the result is decrypted to obtain the final plaintext result. This ensures that sensitive data remains confidential during the entire computation process.

6. ML Privacy Meter

A tool that can automatically assess the privacy risks of machine learning models to their training data can aid practitioners in compliance with data protection regulations. It is a python library that enables quantifying the privacy risks of machine learning models to members in the training dataset and is based on well-established algorithms through membership inference attacks. The tool provides privacy risk scores which help in identifying the data records that are under high risk of being revealed through the model parameters or predictions. The tool can generate extensive privacy reports about the aggregate and individual risk for data records in the training set at multiple levels of access to the model. It can estimate the amount of information that can be revealed through the predictions of a model (referred to as Black-box access) and through both the predictions and parameters of a model (referred to as White-box access). Hence, when providing query access to the model or revealing the entire model, the tool can be used to assess the potential threats to training data.

It considers attackers who can exploit only the predictions of the model, the loss values, and the parameters of the model. For each of the simulated attacks, the tool reports risk scores for all the data records. The larger the gap between the distribution of these scores for records that are in the training set versus records that are not in the training set, the larger is the leakage from the model would be. Success of the attacker can be quantified by an ROC curve representing the trade-off between False Positive Rate and True Positive Rate of the attacker. True positive represents correctly identifying a member as present in the data and False positive refers to identifying a non-member as member. An attack is successful if it can achieve larger values of True Positive rate at small values of False Positive rate. A trivial attack such as random guess can achieve equal True Positive and False Positive Rates. ML Privacy Meter automatically plots the trade-off that are achieved by our simulated attackers. The area under those curves quantities the aggregate privacy risk to the data posed by the model. The higher the area under curve, larger the risk. These numbers not only quantify the success of membership inference attacks, but they can also be seen as a measure of information leakage from the model.

When deploying machine learning models, this quantification of risk can be useful while performing a Data Protection Impact Assessment. The aim of doing a DPIA is to analyse, identify and minimize the potential threats to data. ML privacy meter can guide practitioners in all the three steps. It can help in estimating the privacy risk to data and to identify the potential causes of this risk. It can also be useful in selecting and deploying appropriate risk mitigation measures. The tool produces detailed privacy reports for the training data. It allows comparing the risk across records from different classes in the data. We can also compare the risk posed by providing black box access to the model with the risk due to white box access. As the tool can immediately measure the privacy risks for training data, practitioners can take simple actions such as finetuning their regularization techniques, sub-sampling, re-sampling their data, etc., to reduce the privacy risk. [1]

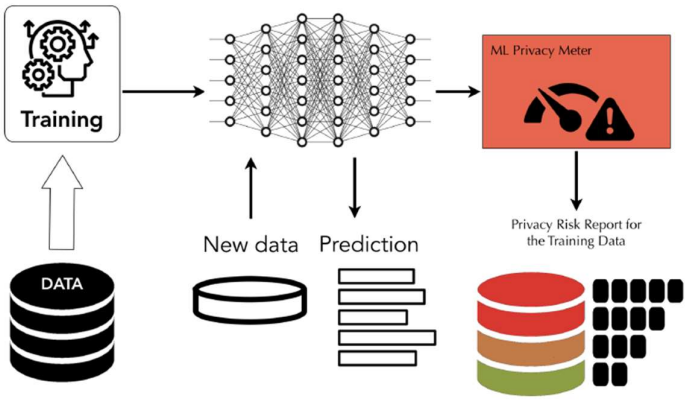

Fig 11: ML Privacy Meter

7. Practical Applications of Privacy-Preserving Machine Learning

7.1 Healthcare

In the healthcare industry, patient data is highly sensitive and must be protected. Privacy-preserving machine learning can be used to develop predictive models for disease diagnosis, drug discovery, and personalized treatment without compromising patient privacy.

7.2 Finance

The financial sector deals with a wealth of sensitive information, such as transaction records, credit scores, and investment portfolios. Privacy-preserving techniques can be used to build fraud detection models, credit risk assessment tools, and personalized investment recommendations while safeguarding customer data.

7.3 Smart Cities

As cities become increasingly connected and rely on data-driven decision-making, privacy-preserving machine learning can play a crucial role in ensuring the privacy of citizens' personal information, such as location data, energy usage, and transportation patterns.

7.4 Biometrics

Biometric authentication, such as fingerprint or facial recognition, is becoming more widespread. Privacy-preserving techniques can be used to protect the sensitive biometric data collected and ensure that it is not misused or accessed by unauthorized parties.[2]

8. Conclusion

The integration of privacy-preserving techniques into machine learning models is crucial for balancing the need for data-driven insights with the imperative to protect individual privacy. This study highlight ed the various threats to data privacy in machine learning, from membership inference attacks to model inversion attacks, and identifies the types of sensitive data most at risk.

Due to privacy concerns, the right to be forgotten, and other legal requirements, users may request the removal of individual data and its influence from machine learning models. Key techniques such as de-identification, differential privacy, homomorphic encryption, and federated learning, etc offer robust solutions to mitigate these risks. De-identification is a widely used approach that involves removing or masking personally identifiable information (PII) from datasets to prevent re-identification while maintaining data utility. By anonymizing data, adding noise, training models on local data, and using advanced encryption methods, we can significantly enhance the privacy and security of machine learning processes. Practical applications in healthcare, finance, smart cities, and biometrics demonstrate the feasibility and effectiveness of these techniques in real-world scenarios.

De-identification is highlighted in several standards as a necessary step when training machine learning models, such as ISO/IEC 20889 which provides guidelines for data de-identification, focusing on techniques to remove or obscure personal identifiers to protect privacy while maintaining data utility.However, these standards often lack detailed elaboration on the specific techniques to be followed or how they should be categorized based on the type of training data, since different techniques cannot be universally applied to every kind of data. It is crucial to recognize that the suitability of de-identification techniques depends significantly on the data type and its intended use.

Moreover, there is a recurring issue where terms like anonymization and de-identification are used interchangeably across various articles and standards. The concern is referred in NISTIR 8053 - De-Identification of Personal Information stating “de-identification,” “redaction,” “pseudonymization,” and “anonymization”. Some authors and publications use the terms “de-identification” and “anonymization” interchangeably. Others use “de-identification” to describe a process and “anonymization” to denote a specific kind of de-identification that cannot be reversed. In some healthcare contexts the terms “de-identification” and “pseudonymization” are treated equivalently, with the term “anonymization” being used to indicate that the mapping pseudonyms to subject identities has been erased. The term “redaction” is sometimes used in a government context to describe the straightforward removal of information that is identifying or otherwise sensitive.[13] Thus, it's important to clarify that anonymization is actually one of the techniques under the broader umbrella of de-identification. Anonymization involves various methods, such as data masking, which serve to protect personally identifiable information (PII) while maintaining data utility. Refer IS/ISO 22857 clause 8.6

Furthermore, although de-identification is often discussed in the context of data privacy, there is no specific standard that focuses exclusively on training data in machine learning. This gap indicates the need for more comprehensive guidelines and standards that address the unique challenges of applying de-identification techniques to training data in machine learning models.

Thus, privacy-preserving machine learning is a crucial aspect of the data-driven world we live in. As the field continues to evolve, it will be essential for data scientists, researchers, and organizations to prioritize the protection of privacy in their data-driven initiatives. The key to unlocking the full

potential of machine learning lies in our ability to balance innovation and data-driven insights while respecting the trust and security of the individuals and organizations we serve.

9. Insights from the industrial visit

During the industrial visit to the National Informatics Centre (NIC) office, we gained valuable insights into de-identification techniques for training data in machine learning (ML). The experts at NIC emphasized that anonymization, which involves the de-identification of data, is a common practice in the industry. They pointed out that personal identifiers typically do not significantly contribute to model performance, so their removal does not harm the efficacy of the ML models. However, it is essential to perform a dependency check by creating a correlation matrix to ensure that no critical dependencies exist between the identifiers and the model output. In cases where an identifier is essential for model performance, careful de-identification is necessary. The most commonly used technique for de-identification involves replacing identifiers and normalizing the data, maintaining the integrity and utility of the training data while ensuring privacy.

Different types of data require specific de-identification techniques. For image data, particularly in the form of DICOM (Digital Imaging and Communications in Medicine) files, all required identifiers are meticulously removed before using the images for training ML models. Text files also undergo specific anonymization techniques to ensure that personal identifiers are effectively removed.

They highlighted the high importance of de-identification for privacy protection. De-identification of data is widely performed in the industry as it is a crucial aspect of privacy. The insights gained underscore the necessity of tailoring de-identification techniques to different types of data to ensure comprehensive privacy protection across various domains.

10. References

1. Murakonda, S.K., & Shokri, R. (2020). ML Privacy Meter: Aiding Regulatory Compliance by Quantifying the Privacy Risks of Machine Learning. *Data Privacy and Trustworthy ML Research Lab, National University of Singapore*.
2. Zhonghong, Privacy-preserving machine learning techniques for protecting sensitive data. *Medium*.https://medium.com/@zhonghong9998/privacy-preserving-machine-learning-techniques-for-protecting-sensitive-data-d199b450e5a9
3. Al-Rubaie, M., & Chang, J. M. Privacy Preserving Machine Learning: Threats and Solutions.
4. Xu, R., Baracaldo, N., & Joshi, J. Privacy-Preserving Machine Learning: Methods, Challenges, and Directions.
5. IS/ISO/IEC 29100 : 2011 Information Technology — Security Techniques — Privacy Framework
6. ISO/IEC 2nd WD 27091
7. [Article]. (n.d.). https://jis-eurasipjournals.springeropen.com/articles/10.1186/s13635-024-00158-3
8. El Mestari, S. Z., Lenzini, G., & Demirci, H. (2023). Preserving data privacy in machine learning systems. *Computers & Security, 137*, 103605.
9. ISO/IEC 20889:2018. Information technology — Privacy enhancing data de-identification terminology and classification of techniques.
10. Google Cloud. (n.d.). Data De-Identification and Privacy. https://www.cloudskillsboost.google/course_templates/1036/video/471652?locale=fr_CA

11. Ganiev, A. A., Kerimov, K. F., & Azizova, Z. I. Understanding of Data De-identification: Issues of Relevance and Problems.Tashkent University of Information Technologies named after Muhammad al-Khwarizmi, Tashkent, Uzbekistan.
12. Tung, Y., Chen, L., Yang, J., & Wang, P. C. Differential Privacy-Based Data De-Identification: Protection and Risk Evaluation System. Department of Communications Engineering, Feng Chia University, Taichung, Taiwan.
13. Garfinkel, S. L. (2015). *NISTIR 8053 - De-Identification of Personal Information*. National Institute of Standards and Technology.
14. ITU-T X.1148 — Supplement on requirements for data de-identification assurance
15. Hengzhu Liu, Ping Xiong, Tianqing Zhu, and Philip S. Yu. 2018. A Survey on Machine Unlearning: Techniques and New Emerged Privacy Risks. J. ACM 37, 4, Article 111 (August 2018), 35 pages.