International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(21s) 2026 pp. 1618-1627
https://doi.org/10.70917/ijcisim-2026-5914
Article

Data Privacy and Security in Artificial Intelligence Models: Emerging Challenges and Defensive Approaches

Ahmad Musa Bulama[1], Mohammad Faisal[2]

[1]Integral University, Dasauli, Bas-ha Kursi Road, Lucknow – 226026 Department of computer Application [0009-0007-5543-8367], ahmadmusa@student.iul.ac.in
[2]Integral University, Dasauli, Bas-ha Kursi Road, Lucknow – 226026 Department of Computer Application [0000-0002-6120-5259], mdfaisal@iul.ac.in

Abstract: The rapid growth of artificial intelligence (AI) and deep learning techniques leads to their extensive use in fields like audio-visual recognition and natural language processing. But there have been recently found some security threats concerning the currently existing mainstream AI models, which might prevent further development of these technologies. In this connection, the problems of data security and privacy protection have become topical for studying. The issues of data leakage and privacy are examined mainly from two points of view: data leakage via model outputs and data leakage via model updates. As for data leakage via model outputs, the article considers principles and state-of-the-art of model theft attacks, model inversion attacks, and membership inference attacks. Data leakage via model updates is considered within the framework of the research of ways to steal private data while performing distributed training. Concerning data and privacy protection, three common ways of defending are considered: model structure defense, information obfuscation defense, and query control defense. In this article, the up-to-date research in the field of data security and privacy protection in AI deep learning models is reviewed. It concentrates on the theoretical background, research results, and practical applications of data theft and defense technologies in AI deep learning models.
Keywords: Artificial Intelligence, Data Security, Privacy Leakage, Privacy Protection

1. Introduction

Artificial Intelligence (AI) technology is advancing rapidly, driven by three key factors: breakthrough progress in deep neural networks (DNN) across various classical machine learning tasks, the maturation of big data processing technologies coupled with the accumulation of vast amounts of data, and significant improvements in hardware computational capabilities. These factors have accelerated the adoption of AI applications in domains such as autonomous driving, image recognition, and speech recognition, thereby facilitating the intelligent transformation of traditional industries. AI technology has already been widely adopted across various sectors in India. In the e-commerce sector, AI is employed for user behavior analysis and network traffic analysis, enabling organizations to efficiently manage high-concurrency tasks while enhancing the robustness of overall systems. In the intelligent transportation sector, AI supports path planning and driver–passenger behavior detection, contributing to improved traffic management and transportation safety. In the financial sector, AI is extensively utilized for high-frequency trading, fraud detection, and anomaly detection, thereby improving operational efficiency and strengthening risk management. In the field of network security, AI facilitates automation testing, significantly enhancing the ability of security personnel to identify anomalies within large-scale datasets. Recognizing the transformative potential of AI, governments worldwide have increasingly incorporated AI into their national development strategies. For instance, in 2017, the Chinese government's work report referenced AI for the first time, elevating its development to the level of a national strategic priority [1].

Copyright: © 2026 by the authors

In most cases, most applications of machine learning require intensive resources. As such, providers of cloud-based services such as Amazon, Google, and Microsoft have introduced machine learning services that cater for the storage and computation needs. The providers provide platforms for training and querying by the service user at the specific instance. Normally, the providers charge the service users according to the number of queries made [2].

However, although the progress in AI technology is very fast, there are still many risks of data leakage in the AI. The parameter of the AI model needs to be guaranteed, because any leakage will bring huge economic loss for model owners. Besides, the sample data for AI usually involves personal privacy, and the leakage will lead to serious economic and legal loss for model owners. The "Cybersecurity Law of the People's Republic of India" was passed in 2017, which stresses the protection of personal privacy information. Hence, to avoid data leakage risk in AI application has become one of the biggest challenges in the development and deployment of AI technology. To guarantee the privacy of the AI model, the cloud service provider keeps the confidentiality of their model by only providing an interface for user access, and thus avoiding user accessing the data of the model directly. However, in recent years, there have been many efforts to compromise the data privacy of AI models. It has been found that when using deep learning models, the computational data of the model, including output vectors, model parameters, and gradients, will leak information about the training data or the attribute of the model. Unfortunately, this data can be inevitably leaked to attackers, especially output vector of the certain models [3].

This creates difficulties when it comes to preventing the data leakage issue in deep learning models. In model inversion attacks, an attacker can discover private data of users from the model output in the reverse engineering process without having access to the data itself. In membership inference attacks, attackers can infer if a certain data sample belongs to the training dataset on the basis of the model output. These attacks can be conducted using the interaction with the cloud service interface only. In practice, such data theft leads to privacy issues including, for instance, recovery of facial images from the output vectors of the face recognition model, which results in leakage of the facial information of users. Model parameter theft from the output results causes great financial damage for the model owner [4].

Also, along with the development of federated learning methods, the attackers may become participants in the process of model training. Although participants in federated learning do not usually have the access to input data from each other, the attackers are able to have access to the output data of models, as well as to the parameters and gradients during the training process. This provides huge opportunities for the attackers to steal the private data of other participants, which makes distributed machine learning a dangerous area. There are many ways which have recently been suggested by many researchers to provide security from privacy attacks on AI technologies. The modification of model structure, special addition of noise to the output vector and using the concept of differential privacy helps to protect models from certain attacks. In this paper, we will consider data inference attacks, which have been broadly studied by researchers: model stealing attack, model inversion attack, and membership inference attack.

2. Data and Privacy Leaks in Artificial Intelligence Systems

The data used in the training and deployment process of the deep learning models is vulnerable to being leaked out. Depending on the model output information that the attackers use to conduct their attack, the attacks may be classified into two categories: model output-based data leakage and gradient update-based data leakage.

2.1. Data Leakage via AI Model Outputs

Model output is defined as the expected outcome obtained by the user after training and deploying the model. For instance, in case of a classification task, the model output will be the expected class or probability vector for each sample. Recent studies have indicated that there is always some data information in the model output results. The model output can be exploited to some extent to obtain the data. There are two main types of data that can be stolen, namely model parameters and training/test data [2].

2.1.1. AI Model Theft

Model theft or extraction refers to a malevolent process whereby the attacker interrogates a black box model to generate outputs, thereby achieving the same functionalities and replicating decision boundaries of the target model. This model has often been built by the owner with huge efforts and investments and is highly valued. When the details of the model have been leaked, the attackers get the chance to use their services for free or avoid paying third-party service fees for commercial gains. However, in an even worse scenario, the attackers would deploy white box adversarial attacks on the online model, making it easy for them to execute a successful attack. This is because when the online AI model has been compromised through model theft attack, white box adversarial examples could be easily created and used to misclassify them. As reported in a study on black-box adversarial attack of online AI classification tasks of Amazon and Google, the success rate of misclassifications was achieved to be 96.19% and 88.94%, respectively [4].

Today, almost all AI service providers provide services where the model is hosted on secure cloud servers, offering paid query services via API interface. Only query samples can be provided through APIs to the model and the results of predictions will be obtained. But even if attackers try to obtain the information from the predicted results, they can steal the model from the server under some conditions by asking queries to the server. Model theft attacks are classified into three categories: Equation solving attack, Meta-model-based model theft, Substitute-model based model theft.

2.1.2. Equation-Solving Attacks on AI Models

Equation Solving Attacks are specifically aimed at classical machine learning algorithms such as SVM. The attacker obtains information about the model, like the type and structure of the algorithm, and creates formulae that can solve for parameters of the model using query responses [5]. Moreover, attackers can steal hyperparameters used in classical algorithms, like loss function weights and regularization factors [6], or parameters like K in K nearest neighbors (KNN) algorithms. This technique needs the attacker to know about the type, structure, and training data of the target algorithm and cannot be used on neural networks.

2.1.3. Meta-Model-Driven Model Extraction Attacks

Model stealing attacks based on the meta-model require training another meta-model $\Phi(\cdot)$ in order to predict certain attributes of the target model. This model accepts the result of processing the task data 'x' by the attacked model and produces predicted attributes such as neural network layers, type of activation functions, etc. In order to train the meta-model, it is required to gather different models with the same functionality of the attacked model, receive the results of their processing of the same datasets, and prepare the training set for the meta-model. However, the preparation of such a dataset is computationally heavy, thus, the attack is quite impractical. For instance, the related experiments were done only on the MNIST digit classification task [7].

2.1.4. Substitute Model-Based Model Theft

The model stealing based on the substitute model is the most feasible attack method at present time. The attacker first asks the target model for a set of samples, gets the predictions, labels the query set by the predictions to generate the training set, and trains a local substitute model with the same task as the target model. Through a long period of training, the substitute model imitates the property of the target model. The attackers prefer to use the deep learning models such as VGG [8] and [9] which are powerful in terms of approximation.

Substitute-model-based model theft is currently the most practical form of attack. The attacker queries the target

model with samples, obtains the prediction results, and uses them to label the query data to create a training dataset.

They then train a substitute model locally with the same task as the target model. After extensive training, the substitute model approximates the target model's characteristics. Attackers often select deep learning models like

VGG [8] and [9] as substitute models with strong fitting capabilities. Unlike Equation-solving Attacks,

substitute-model theft does not require knowledge of the target model's structure; the goal is not to steal the exact parameters but to approximate the target model's functionality. This is possible through querying the model with many samples, although they do not have enough data. Too many queries are costly and might cause the target model owner to suspect the attackers. For this reason, some researchers have suggested using data augmentation methods to improve the query dataset and allow samples to reflect the nature of the target model [4]. The adversarial examples can be used for data augmentation, and this type of sample usually falls around the decision boundary of the model, thus improving the substitute model's performance to mimic the target model [11, 12]. Besides data augmentation, research has shown that using unrelated data to create the dataset leads to attacks with significant impacts, and ways of combining task related and unrelated data have been suggested [2-10].

2.1.5. Privacy Leakage Attacks

However, the results of predictions in machine learning model prediction contain plenty of inference information

about the sample. In different learning tasks, these results have different interpretations. In the case of image

classification tasks, the output of the model is a vector, in which each element stands for the probability of the sample belonging to some class. Recently, it was shown that the results of black-box models can be used to steal information about the training set of the model. [13] proposed the model inversion attack. This method allows us to restore facial images from the training set by using confidence information in the output of the model. The authors used model inversion attacks to analyze common facial recognition models such as softmax regression [14], multilayer perceptron, and autoencoders. The approach demonstrated that confidence information in model outputs contains the information about inputs and can be considered as a criterion for data recovery attacks. Model inversion can be considered as an optimization problem when one tries to minimize the difference between the output vectors of reverse and target data. When the attacker gets the output vector for the particular class, he can apply gradient descent to get the appropriate reverse data.

Membership inference attacks are a more implementable type of attack. Here, the attacker tries to find out if a specific sample belongs to the training data set of the target machine learning model. For instance, an attacker may wish to find out if the data related to a person's health are included in the training data set of the diagnostic machine learning model of some firm. The inclusion of such data may help in determining private details of the person. Data related to the training set of the target machine learning model is known as member data and data not related to the training set is known as non-member data. As the attackers usually do not have access to the target machine learning model, they cannot use these attacks other than in a black-box environment. Membership inference attacks have been very widely studied in literature [15-20]. They have emerged as a main research area in fields such as medical diagnostics and genetic testing.

In 2017, the research work carried out by [15] first introduced membership inference attack techniques. After conducting several experiments, they developed a mechanism for carrying out membership inference attacks in a black-box environment. Membership inference attack works on the principle that the prediction vector for the member samples and the non-member samples are very much different. If the attacker manages to detect this difference, he/she will be able to conduct a membership inference attack. However, when working in a black box environment, only the prediction vectors of the target model are present. In reality, due to certain usage constraints, it is difficult to collect a variety of prediction vectors. Moreover, because the distribution of prediction vectors is inconsistent for the different samples, training with prediction vectors may not yield good attack results. To solve this problem, Shokri et al. used shadow datasets which have a similar distribution as that of the target data set and trained several shadow models for each type of data. By using the prediction vectors, they built an attack model to capture the difference in prediction vectors between member and non-member data, successfully performing membership inference attacks in black-box settings.

2.2. Gradient Leakage Attacks

Gradient updates are referred to as a scenario where the optimization of the model parameters is done through the computation of the gradients during the training process. The gradients, which are computed during training, also tend to possess some information that is privacy-sensitive. Gradient updates usually happen in scenarios of distributed training, where several parties with different datasets each train the model using their own data in each round. The updates are then aggregated for training a common model. In this process, no party, whether it is the central server or any other training party, gets access to the data of other training parties. Yet, while protecting the original data, there may be privacy leakage from gradient updates.

Despite the fact that many approaches have been used in order to ensure the safety of raw data in the training process, during multi-party distributed AI model training, each person uses his or her own data in order to train the current model and sends model parameter updates to other people or a central server. Some of the most recent works that were presented at machine learning and information security conferences have pointed out attacks on model parameter updates that allow getting the information about other parties' training data. For example, [29] used model parameters updated by other users as input features in order to train an attack model that can infer attributes of other users' datasets. Other authors [30,31] used Generative Adversarial Networks (GANs) in order to generate attacks that recover training data of other users. In the case of multi-party collaborative training, a common model is used as a discriminator, and model parameter updates are used as inputs in order to train the generator and extract some training data from a victim's dataset. In the last study [32], the authors did not use GANs; rather, they simply optimized the pixels of simulated images in order to make their gradients similar to gradients obtained during backpropagation in the common model.

3. Data Security and Privacy Protection in Artificial Intelligence

In order to reduce the risks that may be involved in the privacy of model and data leakage during AI training and testing, which may include the leakage of training data caused by parameter update, leakage of model data due to query responses during testing and indirect privacy breach during the normal use of AI models, both academicians and practitioners have tried to find out several solutions from different angles. Without any attack, the information that may be formed through the use of AI models could still result in indirect data privacy leakage. In order to solve the problems of data leakage, the key is to reduce or make obscure the information that could be used without changing the function of the model. The solutions that could be taken include:

- Model Structure Defense: This method entails purposefully tuning the model during training in order to make the model's output less sensitive to different samples.
- Information Obfuscation Defense: This technique entails the obfuscation of information in the interaction data such as the model's output or parameter updates in a manner that preserves model efficiency.
- Query Control Defense: This defense entails monitoring the query operations and rejecting any malicious queries.

3.1. Model Architecture-Based Defense

The model-based defense methods alter the structure of the model to minimize the amount of information leakage or to minimize the model overfitting in order to protect the model and the data leakage. In their research work, [33] found out that when the target model is a decision tree, CART decision tree variations could be used to modify the priority of sensitive features. From their experiments, they realized that placing sensitive attributes at the root or leaf nodes provided significant protection against the attack. When sensitive attributes are placed at the root node, the maximum protection is achieved.

In their studies, [15] and [17] proposed introducing Dropout layer to the target model, model stacking for combining several meta-learners, and addition of regularization terms. From their experiments, they confirmed that these techniques minimized the attack accuracy in member inference attacks. [34] proposed a method of defense that

uses adversarial learning where once the success rate of resistance to member inference attacks is estimated and added to the loss function as an adversarial regularization term, then adversarial training with MIN-MAX approach could yield a model that minimizes the success rate of the attacks while maximizing the accuracy of classification.

Moreover, [35] proposed MIASec, which changes the properties of the training data in the target model to make it difficult to differentiate between prediction vectors of member and non-member data. As described earlier, the basis of model inversion attacks is the information present in the output vectors, while the member inference attacks depend on the uneven distribution of prediction vectors in training and test samples.

Thus, to protect the model from model inversion attacks, it is necessary to minimize the relationship between output and input vectors, while member inference attack protection requires minimization of distribution differences of output vectors. In model-based protection, the goal is to change the architecture and loss function of the model to minimize the information in output vectors, which provides strong protection of the model.

However, this technique is not without flaws since it can significantly affect the work of the model, leading to classification accuracy fluctuations. In this case, the defender needs to find the right balance between the performance and robustness of the model.

Recently, researchers began combining machine learning and cryptographic approaches to ensure model privacy. [36] suggested using techniques of differential privacy to change gradients in distributed training in order to protect the privacy of the training dataset. While it can lead to reduction in performance of the model in the end, it greatly increases the privacy of the training dataset. [37] used secure multi-party computation technology to implement encrypted matrix multiplication and activation functions, thereby protecting the privacy of the training dataset during training. These techniques of machine learning for data privacy protection can also be used to defend against data leakage, increasing privacy of training datasets of models.

3.2. Information Obfuscation-Based Defense

Defenses oriented towards data include obfuscation strategies that apply to either the input data set or the predicted output of the model. These obfuscations seek to interfere with the informative nature of the output generated by the AI model without compromising the correctness of the output. The main types of data obfuscations include truncation obfuscation and noise obfuscation.

The approach to truncation obfuscation was presented by [15], where the idea was that the output vector can be truncated based on the target model's generation, such as showing only the results for the categories with higher probability values or decreasing the number of decimal places in the output vector. The approach to rounding the output vector was introduced by [33], providing a similar result. Mem-Guard uses the idea of adversarial samples for noise obfuscation proposed by [38]. They found out that attacks based on member inference are highly vulnerable to modifications of the prediction vector provided by the target model. It is possible to add some special noise to the prediction vector to confuse the distribution of difference between the prediction vectors of member and non-member data; as a result, a "noise obfuscation sample" is created which does not impact the actual result and protects from the attacks. The differential privacy method was used for obfuscation in the paper [39]; it was proposed to add some noise to the output vector. It was suggested that using the differential privacy algorithm will allow removing the features from the output vector while keeping the classification information. Moreover, it was proposed to add some noise to the loss function; as a result, some decrease in classification accuracy occurs but the output vector becomes more private and the attack can be prevented. Both the mentioned types of attacks (model inversion and member inference) use the output vector of the target model. Thus, if it is possible to change the output vector explicitly without changing the result, it is possible to disrupt the effective information stored there and protect from these attacks. But this method is limited. Minimal changes in the output vector do not provide high protection, and extensive changes make the vector unusable for classification. Thus, some compromise between privacy and usability should be reached.

3.3. Query Control-Based Defense

Query control defense can be understood as a scenario in which the defender derives features from user query behavior to protect themselves from the privacy leakage attack. In order to launch a privacy leakage attack, the attacker will have to ask many queries on the target model and in some cases manipulate their input vectors for faster attack. Studying the user query behavior features allows you to detect attackers and block their queries to prevent the attack.

In the process of anomaly sample detection, attackers have to ask many questions in order to acquire the black-box online model. In order to improve efficiency, attackers will deliberately change normal samples. In response to the attack on model leakage, defenders try to detect anomalies in queries about anomalous samples in order to detect theft behavior in the model. PRADA [2] is a defense technology for detecting model theft attacks. It determines if the user is trying to steal the model according to the distribution of the distance between multi-sample features. The study has found that the distribution of the distance between the features of randomly selected normal samples approximates a normal distribution. On the contrary, samples that are queried for model theft will be deliberately changed, so the distribution of the distance between their features will be quite different from the normal distribution. Multiple queries can be tested for abnormalities in users. Feature distribution of query samples can be used for detection.[40] registered users' query samples and analyzed their feature distribution in order to determine the possibility of model theft. [12] pointed out that the feature distribution of normal samples and artificially modified samples is quite different.

Due to the large numbers of queries performed by attackers when performing the attack in the query behavior detection, their query behavior would differ significantly from the regular users. This can help defend against model and data leakage attacks to a certain extent. [39] proposed defending against member inference attacks through the characteristic of user query behavior during the sample input process to defend against data leakage attacks. When the attackers perform the member inference attacks, they might need to perform multiple queries to the target model. Model providers could limit the number of queries according to the frequency of the user queries, making it more expensive for attackers to carry out member inference attacks.

As discussed above, defenders could defend against model and data leakage attacks through detecting the anomalous samples and queries. However, this type of defense is not very precise and would lead to a relatively high misclassification rate. Query control defense mainly functions when the attack model is being trained and cannot function against the pre-trained attack models. Also, if the attackers know that the target model utilizes the query control defense, they can use various ways to bypass this defense mechanism, including constructing harder-to-detect anomalous samples and using virtual IP addresses.

4. Future Research Directions

4.1. Development of Efficient Data Leakage Attack Techniques

The fundamental principle of data leakage attacks is that model parameters, prediction outputs, and intermediate representations are generated from input data. Consequently, these artifacts inherently encode information about the original training samples, making all artificial intelligence (AI) models susceptible to some degree of data leakage. As a result, completely eliminating the risk of data leakage remains a significant challenge. Future research on data leakage attacks is expected to focus on two primary directions. First, researchers will continue to optimize attack models to improve their capability to infer and reconstruct sensitive information from model outputs. Second, data leakage techniques will be extended to emerging AI paradigms, including transfer learning, reinforcement learning, federated learning, and other collaborative learning environments, where new attack surfaces continue to emerge.

In addition, privacy theft attacks based on model outputs typically require a large number of interactions with the target model. For example, model extraction attacks often require thousands of queries to train an effective substitute model because of the high-dimensional parameter space, nonlinear behavior, and non-convex optimization characteristics of deep neural networks (DNNs) [10]. Such extensive querying not only increases the computational cost of the attack but also raises the likelihood of detection by defense mechanisms. Consequently, improving the efficiency of privacy theft attacks has become a major research objective. Recent studies have proposed various sample selection strategies that utilize highly informative and representative queries to maximize information extraction while

minimizing the number of model interactions [41,42]. These strategies include active learning techniques [43,44], natural evolution strategies, and other query optimization approaches. Although these advances enhance the effectiveness of adversarial attacks, they also provide valuable insights for designing stronger privacy-preserving mechanisms. Ultimately, continued research into data leakage attacks and corresponding defense techniques will contribute to the development of more secure, robust, and privacy-aware AI systems capable of resisting increasingly sophisticated threats.

4.2. Development of Effective Defense Mechanisms Against Data Leakage Attacks From the above discussion, data leakage attacks occur as a result of construction or inference results from the output of a model which inherently carries specific private data. Defenses against data leakage attacks therefore can mainly evolve in the following three aspects:

• Obfuscate the output vector by minimizing the information that it contains.

• Obfuscate the private data by adding specific noise in the private data so as to minimize the information that the inference results of the model carry.

• Obfuscate the parameters of the model through privacy preserving machine learning techniques where the internal parameters, intermediary results, and output vectors are encrypted to minimize the information leakage. It is however important to consider the extent of the obfuscation of different pieces of information in building up defenses against data leakage attacks. Failure to obfuscate sufficiently will mean that the attackers can still obtain private information. Too much obfuscation will also affect the usability of the model and damage its fundamental operation. Similar problems also come out in implementing other defense technologies such as query control defenses where stringent query control rules will effectively stop privacy data leaks but make it difficult for the regular user and might end up wrongly labeling the legitimate user as an attacker. In order to ensure privacy data obfuscation while making sure the model provides its fundamental services, privacy data leakage defense technologies need to have an effective balance between security and model usability.

5. Conclusion

Artificial intelligence (AI) is widely used in healthcare, finance, transportation, education, and other critical sectors, making data security and privacy major research concerns. This paper reviewed recent advances in AI data security, focusing on data leakage attacks and corresponding defense mechanisms. Common attacks include model inversion, membership inference, model extraction, and gradient-based attacks in federated learning, all of which can expose sensitive training data even without direct dataset access.

To counter these threats, researchers have proposed model structure defenses, information obfuscation techniques such as differential privacy and encryption, and query control mechanisms that limit malicious access. Emerging approaches also incorporate adversarial training, secure aggregation, homomorphic encryption, and secure multi-party computation to strengthen privacy protection.

Despite these advances, significant challenges remain. The black-box nature of deep learning limits model interpretability, making it difficult to identify and prevent information leakage. Additionally, many defense methods introduce trade-offs between privacy, computational efficiency, and model accuracy, particularly in resource-constrained environments. The rise of large language models (LLMs) and multimodal AI has further introduced new privacy risks, including prompt leakage, training data memorization, and advanced model extraction attacks.

Future research should focus on developing interpretable AI models, adaptive defense mechanisms, efficient privacy-preserving learning methods, and standardized evaluation frameworks. Integrating explainable AI, zero-trust architectures, blockchain auditing, confidential computing, and advanced cryptographic techniques can further enhance AI security. Overall, ensuring secure, trustworthy, and privacy-preserving AI systems remains an open challenge that requires interdisciplinary collaboration to support the safe deployment of AI across critical applications.

ACKNOWLEDGMENT

The authors gratefully acknowledge the support of the Department of Computer Application, Integral University, Lucknow, India. This manuscript has been assigned the Institutional Manuscript Communication Number (MCN): IU/R&D//2026-MCN0004847.

REFERENCES

1. C. Tumma, R. Azmeera, S. Ayyamgari, and B. Y. R. Thumma, "Data security and privacy protection in artificial intelligence models: Challenges and defense mechanisms," Int. J. Sci. Res. Eng. Manage., vol. 9, no. 1, pp. 1–9, 2025.
2. S. Sankaran, “Ensuring trust and security in AI-powered chatbots: A framework for standardization,” in Proc. 2025 IEEE Conf. Artif. Intell. (CAI), May 2025, pp. 1627–1632.
3. S. Wassan, Liudajun, H. Ying, H. Dongyan, and P. Fei, "Federated learning and differential privacy: Machine learning and deep learning for biomedical image data classification," Digital Health, vol. 11, Art. no. 20552076251358531, 2025.
4. A. R. Yadulla, V. K. Kasula, M. Yenugula, and B. Konda, "Enhancing cybersecurity with AI: Implementing a deep learning-based intrusion detection system using convolutional neural networks," European Journal of Advances in Engineering and Technology, vol. 10, no. 12, pp. 89–98, 2023.
5. K. Zhang, X. Yuan, R. Sun, C. Hong, and J. Xue, "Traceable and collision-resilient differential privacy," IEEE Transactions on Information Forensics and Security, vol. 20, pp. 11816–11829, 2025.
6. X. Zhang and Q. Zhang, "Defending against attacks in deep learning with differential privacy: A survey," Artificial Intelligence Review, vol. 58, no. 11, Art. no. 347, 2025.
7. A. Al-Dujaili and U. M. O'Reilly, "Sign bits are all you need for black-box attacks," in International Conference on Learning Representations (ICLR), Apr. 2020.
8. L. Demelius, R. Kern, and A. Trügler, "Recent advances of differential privacy in centralized deep learning: A systematic survey," ACM Computing Surveys, vol. 57, no. 6, pp. 1–28, 2025.
9. L. Li, S. Zhang, and B. Wang, "Plant disease detection and classification by deep learning—A review," IEEE Access, vol. 9, pp. 56683–56698, 2021.
10. B. Wang, C. Yang, and J. Ma, "UKVLDP: Utility-optimized local differential privacy mechanism for key–value IoT data collection," IEEE Internet of Things Journal, vol. 12, no. 11, pp. 17964–17976, 2025.
11. P. M. T. Do, J. Lu, Q. Zhang, and G. Zhang, "A federated graph neural network with differential privacy for cross-domain recommender systems," ACM Transactions on Intelligent Systems and Technology, vol. 16, no. 4, pp. 1–25, 2025.
12. H. Yu, K. Yang, T. Zhang, Y. Y. Tsai, T. Y. Ho, and Y. Jin, "CloudLeak: Large-scale deep learning models stealing through adversarial examples," in Proc. Network and Distributed System Security Symposium (NDSS), Feb. 2020.
13. M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in Proc. 22nd ACM SIGSAC Conf. Computer and Communications Security (CCS), Oct. 2015, pp. 1322–1333.
14. V. K. Kasula, "AI-driven banking: A review on transforming the financial sector," World Journal of Advanced Research and Reviews, vol. 20, no. 2, pp. 1461–1465, 2023, doi: 10.30574/wjarr.2023.20.2.2253.
15. R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in Proc. IEEE Symp. Security and Privacy (SP), May 2017, pp. 3–18.
16. S. Yeom, I. Giacomelli, M. Fredrikson, and S. Jha, "Privacy risk in machine learning: Analyzing the connection to overfitting," in Proc. IEEE 31st Computer Security Foundations Symposium (CSF), Jul. 2018, pp. 268–282.
17. A. Salem, Y. Zhang, M. Humbert, P. Berrang, M. Fritz, and M. Backes, "ML-Leaks: Model and data independent membership inference attacks and defenses on machine learning models," in Proc. 26th Annual Network and Distributed System Security Symposium (NDSS), San Diego, CA, USA, Feb. 24–27, 2019.
18. Y. Long, V. Bindschaedler, and C. A. Gunter, "Towards measuring membership privacy," arXiv preprint arXiv:1712.09136, 2017.
19. Y. Long, V. Bindschaedler, L. Wang, D. Bu, X. Wang, H. Tang, et al., "Understanding membership inferences on well-generalized learning models," arXiv preprint arXiv:1802.04889, 2018.
20. S. Yeom, M. Fredrikson, and S. Jha, "The unintended consequences of overfitting: Training data inference attacks," arXiv preprint arXiv:1709.01604, 2017.
21. D. Babu Sam, S. Surya, and R. Venkatesh Babu, "Switching convolutional neural network for crowd counting," in Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), 2017, pp. 5744–5752.
22. J. Kuha and C. Mills, "On group comparisons with logistic regression models," Sociological Methods & Research, vol. 49, no. 2, pp. 498–525, 2020.
23. Pal, M. (2005). Random forest classifier for remote sensing classification. International journal of remote sensing, 26(1), 217-222.
24. L. Song, R. Shokri, and P. Mittal, "Privacy risks of securing machine learning models against adversarial examples," in Proc. ACM SIGSAC Conf. Computer and Communications Security (CCS), Nov. 2019, pp. 241–257.
25. A. Salem, A. Bhattacharya, M. Backes, M. Fritz, and Y. Zhang, "Updates-Leak: Data set inference and reconstruction attacks in online learning," in Proc. 29th USENIX Security Symposium (USENIX Security 20), 2020, pp. 1291–1308.
26. J. Hayes, L. Melis, G. Danezis, and E. De Cristofaro, "LOGAN: Membership inference attacks against generative models," Proc. Privacy Enhancing Technologies (PoPETS), 2019.

27. M. Nasr, R. Shokri, and A. Houmansadr, "Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning," in Proc. IEEE Symp. Security and Privacy (SP), May 2019, pp. 739–753.
28. K. Leino and M. Fredrikson, "Stolen memories: Leveraging model memorization for calibrated white-box membership inference," in Proc. 29th USENIX Security Symposium (USENIX Security 20), 2020, pp. 1605–1622.
29. L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," arXiv preprint arXiv:1805.04049, 2018.
30. Z. Wang, M. Song, Z. Zhang, Y. Song, Q. Wang, and H. Qi, "Beyond inferring class representatives: User-level privacy leakage from federated learning," in Proc. IEEE Conf. Computer Communications (INFOCOM), Apr. 2019, pp. 2512–2520.
31. B. Konda, V. K. Kasula, M. Yenugula, A. R. Yadulla, and S. R. Addula, "Homomorphic encryption and federated attribute-based multi-factor access control for secure cloud services in integrated space-ground information networks," International Journal of Communication and Information Technology, vol. 3, no. 2, pp. 33–40, 2022.
32. L. Zhu, Z. Liu, and S. Han, "Deep leakage from gradients," in Advances in Neural Information Processing Systems (NeurIPS), vol. 32, 2019.
33. R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in Proc. IEEE Symp. Security and Privacy (SP), May 2017, pp. 3–18.
34. S. Shaham, A. Hajisafi, M. K. Quan, D. C. Nguyen, B. Krishnamachari, C. Peris, et al., "Privacy and fairness in machine learning: A survey," IEEE Transactions on Artificial Intelligence, vol. 6, no. 7, pp. 1706–1726, 2025.
35. H. Zhao, C. Pan, M. Liu, D. Jiao, and H. Zhu, "Adaptive noise trap of layered differential privacy against privacy leakage for large-scale industrial federated learning," IEEE Transactions on Network Science and Engineering, vol. 12, no. 5, pp. 4044–4059, 2025.
36. A. K. Abasi, M. Aloqaily, and M. Guizani, "6G mmWave security advancements through federated learning and differential privacy," IEEE Transactions on Network and Service Management, vol. 22, no. 2, pp. 1911–1928, 2025.
37. L. Yang, M. Tian, D. Xin, Q. Cheng, and J. Zheng, "AI-driven anonymization: Protecting personal data privacy while leveraging machine learning," arXiv preprint arXiv:2402.17191, 2024.
38. L. d'Aliberti, E. Gronberg, and J. Kovba, "Privacy-enhancing technologies for artificial intelligence-enabled systems," arXiv preprint arXiv:2404.03509, 2024.
39. L. Cohen, K. Wang, B. Kang, and S. Mannor, "Improving token-based world models with parallel observation prediction," arXiv preprint arXiv:2402.05643, 2024.
40. C. Chen, L. Wei, J. Xie, and Y. Shi, "Privacy-preserving machine learning based on cryptography: A survey," ACM Transactions on Knowledge Discovery from Data, vol. 19, no. 4, pp. 1–33, 2025.
41. M. Mohammadi, M. Vejdanihemmat, M. Lotfinia, M. Rusu, D. Truhn, A. Maier, and S. Tayebi Arasteh, "Differential privacy for medical deep learning: Methods, tradeoffs, and deployment implications," npj Digital Medicine, vol. 9, no. 1, Art. no. 93, 2026.
42. S. Zhu, X. Yuan, B. Ma, W. Ni, and W. Zhang, "Beyond spatial privacy: Protecting trajectories with spatio-temporal differential privacy," IEEE Transactions on Knowledge and Data Engineering, 2026, early access.
43. E. N. Kucur, T. Buyuktanir, M. Ugurelli, and K. Yildiz, "Privacy-preserving machine learning techniques: Cryptographic approaches, challenges, and future directions," Applied Sciences, vol. 16, no. 1, Art. no. 277, 2025.
44. A. M. Bulama and M. Faisal, “Analysis and design of a framework handling information security in AI-powered service bots,” International Journal of Computer Information Systems and Industrial Management Applications, vol. 18, no. 15S, pp. 1–9, 2026.