Security and Privacy in Mobile ML Pipelines

Dheeraj Vaddepally

dheeraj.vaddepally@gmail.com

Abstract

The emergence of the mobile applications that are driven by machine learning (ML) has brought a breakthrough in the world of healthcare, e-commerce, and entertainment with its personalized and smart user experiences. Despite the wide application of ML pipelines in mobile environments, the security and privacy issues that could be raised have become an issue of high concern on a global scale. The storage and processing of data, combined with the deployment of advanced ML models, expose vulnerabilities to risks such as data leakage, adversarial attacks, and unauthorized access. These threats not only compromise user trust but also incur applications with financial and brand damage.

The present article deals with the important problems of the safety of mobile ML pipelines, namely adversarial manipulations, insecure storage, and transmitting of data. The vital techniques that can help to work out those problems are described along with the most popular ones, namely secure model storage based on encryption, federated learning for data transfer reduction, and model robustness tactics that will increase adversarial attacks defense. Besides that, new technologies like homomorphic encryption and blockchain are also discussed as ways to secure model updates. Consequently, this paper is involved in the technical and practical levels of the project, and it is meant to show the importance of building secure and privacy-preserving ML pipelines that do not only keep data for user trust but also maintain efficiency. It serves the purpose of opening the way to moving to secure mobile ML solutions in the time of intelligent mobile systems that have started to be of great interest.

Keywords: Mobile application, Machine learning, Models, Pipelines, Blockchain, security

I. INTRODUCTION

With the inception of machine learning, mobile apps have become the intelligent systems that can understand users' personalities and provide them with the most interactive user experience. Nowadays, from virtual assistants like Siri and Google Assistant to recommendation engines in e-commerce platforms and diagnostic tools in healthcare, mobile ML pipelines are an inseparable part of modern apps. These pipelines cover various stages of development such as data collection, preprocessing, model training, the deployment of models, and inference helping it to work and be more accurate and efficient[1].

The extended reliance on the user data which is sensitive for training and inference makes clear the importance of security and privacy in mobile ML pipelines. It applies the personal information on location, browsing history, biometric statistics, and health indicators to formulate the user experience. This kind of increased dependence has a higher risk, considering the fact that most mobile devices are used in resource-limited environments and get connected through networks with weak security layers prone to breach[2].

However, promising they are, mobile ML pipelines also bring challenges that need to be overcome. Among these are data leakage and manipulative attacks that are the two top ones. Data breaches are significant risks as they might expose the data of the particular user, while manipulative attacks may make

the models generate false or potentially dangerous data. Furthermore, insecure storage and transmission of ML models can lead to unauthorized access and tampering[3].

Such problems can be solved with a double perspective: strong security while retaining the efficiency and usability of mobile applications. Finding the balance point is tricky because of devices' limitations, such as limited computing capacity and the need to save as much energy as possible[1].

This article is about the potential dangers that come with locking down mobile ML pipelines and a potential method for mitigating them in a secure way. In particular, three main risks are covered including data leakage, adversarial attacks, and insecure model handling. The problem in terms of the security of storing and transmitting the data by using different techniques, including encryption, federated learning, and adversarial training, will be analyzed in the paper.

II. LITERATURE REVIEW

A. *Mobile ML Pipelines Overview*

Mobile machine learning (ML) pipelines form the backbone of many modern mobile applications, enabling intelligent and context-aware services. These pipelines consist of several key stages:

- Data Collection: Raw data, often sensitive, is gathered from users via sensors, apps, and system logs.
- Data Preprocessing: Data is cleaned, normalized, and formatted to suit ML model requirements[2].
- Model Training: Models are trained using datasets, which may include sensitive user information.
- Model Deployment: Trained models are optimized and deployed on devices or servers for real-time inference.
- Inference: Models generate predictions or actions based on user input and real-time data[3].

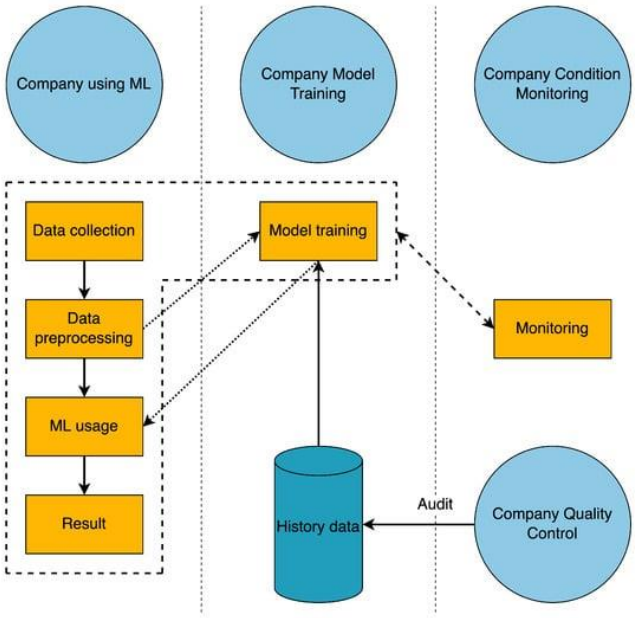

Fig 1. ML pipelines and stakeholders [4].

There are particular weaknesses attached to each of the development steps. Among the period of data gathering, unencrypted communication channels can pave the way to data leaks. Distorting pipelines allows for the embedding of poisoned data, leading to biased models that the rest of the process will examine. The privacy of model training can be interrupted: sensitive training data may be accidentally leaked. Deployment

and inference stages hold threats of model theft, tampering, and adversarial attacks, by which the application could be incapacitated and users' trust could be breached[3].

Attack Against	Attack Type	Caused Phase of ML	Impact	Reference
Availability	Dos Attacks	ML model	Triggers a buffer overflow in image processing	[36]
Confidentiality	Data Reconstruction Attacks	ML training	Modify data	[11,37]
Confidentiality	Attribute Inference Attacks	ML training	Inferring the value of users' private properties	[46]
Confidentiality	Membership Inference Attacks	ML training	Learning specific data	[13]
Integrity	Adds calibrated noises	Data Collection, ML training	Exploit training/inference result	[42,43,44,47]
Accountability	Model stealing	ML model	Learn the model	[45,48]

Fig 2. Overview of security challenges in industrial machine learning pipelines[4].

B. Risks of Data Leakage and Adversarial Attacks

The increasing reliance on sensitive user data, together with the usage of cloud-based and direct-to-device ML processing, has led to a situation where mobile ML pipelines are becoming the main prey for security breaches. If there is insufficient encryption, insecure APIs, insiders who are threats, and data that is not properly protected, the data leakage will be imminent. For example, high-profile breaches such as those of the healthcare apps, where the violations of privacy have resulted in the exposure of private medical records[4].

The most important aggression, of course, is related to the creation of adversarial inputs by malicious actors, which leads to the deception of the ML models. In the area of mobiles, adversarial examples can be the cause of virtual assistants' misinterpretation of voice commands and recommendation systems suggesting a person's learning of inappropriate content. Research studies like the one by Goodfellow et al. (2015) show that the neural networks are one potential target for adversarial perturbations which can hardly be detected by humans[5].

One hears frequently about the grave risks associated with data breaches. For example, a famous fitness app in 2021 leaked the user location data accidentally while the AI apps such as the facial recognition system were under the radar for being infiltrated by adversarial attacks that exploit biases and vulnerabilities. These incidents underscore the pressing nature of forces requiring strong safety measures[4][6].

C. Techniques for Secure Storage and Transmission

Several techniques have been proposed to secure mobile ML pipelines:

1. Encryption Standards: Advanced encryption techniques, such as AES (Advanced Encryption Standard) and RSA (Rivest–Shamir–Adleman), protect data at rest and during transmission. However, these methods can introduce computational overhead, making them challenging to implement on resource-constrained devices[7].

2. Federated Learning: This approach trains ML models across decentralized devices without transferring raw data to a central server. While federated learning reduces the risk of data leakage, it remains vulnerable to model poisoning attacks and requires efficient communication protocols to handle updates.

3. Model Compression: Techniques like quantization and pruning reduce model size, enabling secure on-device storage. These methods also enhance energy efficiency, but they may compromise model accuracy if not implemented carefully[8].

D. Limitations in Existing Research

However, even with so much progress being made, there are still many gaps to secure mobile ML pipelines:

- Security vs. Performance Trade-offs: cryptography and federated learning engage computational and communication costs leading to a real-time mobile computing application which requires a lot of work to be able to use them[1].
- Adversarial Robustness: Mobile environments often see weak defenses against adversarial attacks, because shooting in real-time to create novel attacks is simpler for their authors.
- Bulletproof Model Deployment: Proving the correctness of the update process of the model during the production phase is almost always neglected, what to speak of the adversarial cases.

Prior research has laid the groundwork for safe mobile ML pipelines, but these challenges cannot be tackled completely by that. This paper is aimed at surmounting these obstacles by introducing innovative techniques and practical issues in realizing ML privacy-preserving and robust systems for smartphones[4].

III. Risks of Data Leakage and Adversarial Attacks

A. *Data Leakage Risks*

Data leakage is the most serious fault in mobile ML platforms since they usually manage user's sensitive data that may include location data, personal identifiers, health measures, and browsing history. Data leakage is happening when strangers find their way into the sensitive data system, which is not supposed to be theirs, either during transmission, processing, or storage.

Weak Encryption: Inadequate encryption techniques or unencrypted data transmissions make it easier for attackers to intercept and exploit sensitive information[6].

IV. Risks of Data Leakage and Adversarial Attacks

A. *Data Leakage Risks*

Data leakage is the most serious fault in mobile ML platforms since they usually manage user's sensitive data that may include location data, personal identifiers, health measures, and browsing history. Data leakage is happening when strangers find their way into the sensitive data system, which is not supposed to be theirs, either during transmission, processing, or storage[8].

Weak Encryption: Inadequate encryption techniques or unencrypted data transmissions make it easier for attackers to intercept and exploit sensitive information. For example, applications which use obsolete encryption standards, like SSL rather than TLS, are of course highly vulnerable to man-in-the-middle (MITM) attacks[8][4].

Insecure APIs: APIs form an indispensable part of mobile ML applications for data exchange between clients and servers. In that regard, poorly secured APIs due to weak authentication or excessive permissions can expose critical data to unauthorized access[9].

Lack of Proper Protection of Sensitive Data: Data is often kept on devices or servers with poor encryption or strong access controls, thus increasing the possibility of leakage from insider threats or unintentional exposure.

B. *Adversarial Attacks*

Adversarial attacks are malicious attempts to take advantage of vulnerabilities in ML models. To quote in the context of mobile ML applications, adversarial examples are crafted inputs designed to deceive models into creating wrong or harmful outputs[10].

- Types of Attacks

Evasion Attacks. In this form of attack, adversaries craft their inputs during the inference phase to bypass model defenses. For instance, evasion techniques are used to influence facial recognition systems or virtual assistants to execute unintended commands.

Model Inversion. Adversaries extract sensitive training data such as biometric information or medical records in this attack[7].

Data Poisoning: Attacker injects corrupt or biased data into the dataset during the training phase, which is used to train a model whose output might be corrupted[11].

C. *Adversarial Attacks on Mobile ML Applications*

Adversarial attacks are the most serious threats to mobile applications:

Security Breach: Adversarial examples can manipulate an authentication mechanism, like biometric authentication, to obtain unauthorized access.

When the attack generates suggestions, it may result in negative recommendations for e-commerce or recommendation systems, as well as a failure to personalize the service, which might interrupt services.

Misinformation: The virtual assistants may be readily manipulated to transmit false information or behave in ways that are not intended by malicious voice commands or text inputs[5].

D. *Impact on User Privacy and Security*

Data leakage and adversarial attacks have an impact that extends well beyond technical concerns and has dramatic effects on end-users.

Identity Theft: The leaked information can be exploited by malicious users for identity theft, which might lead to financial loss or fraud[12].

Erosion of Trust: Recurring data breaches or model flaws may result in consumers losing confidence in mobile applications and their suppliers. The erosion of confidence may have enduring consequences for the acceptance of AI-driven technology[12][9].

Financial Risks. The financial burden for companies affected by data breaches includes legal penalties and compensations with reputational loss. For the end-users, there may also be financial consequences when the critical applications, such as banking or healthcare platform, are damaged by adversarial attacks[6].

E. *Techniques for Safe Model Storage and Transmission*

One of the best ways to ensure that models and data are treated properly in mobile machine learning (ML) pipelines is through the secure storage and transmission that protect user information that is confidential but allow the system to be intact. It is not enough to develop mobile device technologies that are also powerful, but to make them also secret, the so-called "protected" models always must be in place. Consequently, such technologies will prevent such threats that include hacking. They are showing up quite frequently every day[8].

Dependable Model Storage on Mobile Devices.

It inhibits unauthorized access or alteration, which might compromise the security and integrity of essential data.

Methods for Dependable Model Preservation

Importance of Encryption

- Encryption has emerged as the predominant method for safeguarding model data on devices.
- AES (Advanced Encryption Standard): AES is widely used for encryption purposes due to its efficiency and strength.
- ML models are going to be kept in mobile phones; any of the models can be executed using AES-256 so that these models cannot be accessed through brute force attack.

- To decrypt models protected by Advanced Encryption Standard (AES), RSA is employed to encrypt smaller data blocks, such keys for symmetric encryption techniques[7].

Security that Relies on Hardware

- These are also reinforced by systems that safeguard models.
- The TEE isolates the data so that the models and calculations may be stored securely.
- Examples of TEEs include Apple's safe Enclave and ARM's Trust-Zone.
- Models in TEEs are immune to malware or unauthorized access in a typical OS.
- Secure Boot and Code Signing: Secure Boot ensures that a firmware and a device's OS is authenticated prior to execution such that no sort of model storage will be ever accessed[3].

Access Control and Authentication

- Stored models should be access-restricted using robust authentication mechanisms.
- Biometric Authentication: Fingerprint or facial recognition ensures that only authorized users have access to sensitive ML models.
- Role-Based Access Control (RBAC): This is assigning roles to users. It limits the access of the model based on predefined permissions and reduces exposure to insider threats[8].

F. Techniques for Secure Transmission

During the transfer of models or data between devices and servers, strong encryption and privacy-preserving techniques must be used to avoid interception and tampering.

Transport Layer Security(TLS)

- Models and information can be guaranteed to be sent via encrypted routes. Among its advantages:
- Protects against man-in-the-middle attacks
- Assures the correct and safe transfer of encryption credentials during ongoing interactions[7].

Differential Privacy

Anonymizes the sensitive data by differential privacy, so it has the resistance property to inference attacks while transmitting

- Randomized Noise Addition: It adds small amounts of noise to the data transmitted so that the individual records of a user cannot be traced in the aggregate datasets.
- Differential confidentiality is advantageous for operations such as tailored suggestions, where private user information is transferred to central facilities[8].

Confirmation of Information Integrity

Hash-based methodologies, such SHA-256, ensure that transmitted data remains unaltered, hence providing confidence to the pipeline [8].

Sophisticated Techniques

Advanced ML technologies address both issues of security and efficiency. Many such techniques apply particularly to resource-constrained computing platforms like smartphones.

1. Federated Learning

Federated learning allows devices to jointly train a model without ever transmitting raw data to central servers. This includes data privacy, whereby sensitive user data is kept on devices, thereby reducing the risk of leakage[7].

Secure Aggregation: Aggregation of updates from devices is done securely with encryption, and thus individual contributions cannot be extracted.

Applications: Federated learning has been used by Google for Gboard keyboard personalization without violating the privacy of the users[11].

2. Model Compression

Pruning and quantization techniques reduce the size of the model but improve security.

Pruning: Pruning simply removes the redundant weights in a neural network that reduces the attack surface against adversarial inputs.
Quantization: It reduces storage by converting the model weights into lower precision (e.g., from 32-bit to 8-bit), making models less interpretable to attackers.
Security Benefit: Compressed models are less susceptible to reverse engineering due to their reduced complexity[11].

Attacks on Pipeline	Countermeasures
Spoofing, Tampering on Data Collection	TLS is used by the system to read data from the server and to confirm the server's authenticity. Audit trails in the blockchain mitigates identity frauds and therefore data manipulation.
Tampering, Elevation of Privilege on Pre-Processing	Validate TLS use and data certificate verification. The blockchain ensures tamper resistance.
Tampering, Repudiation on ML Model	The model was distributedly stored on IPFS. Keeping the model's certificates and hash in the blockchain ensures tamper resistance.

Fig 3. Relevant attack and countermeasures [4].

V. FUTURE SCOPE

The applications at the moment use machine learning (ML) which has improved the efficiency and has a better performance in this manner because it targets clients not on the basis of the required content only - on the contrary, they create and transmit to the customers the specific contents that are the most interesting to them, and besides, they do it on an individualized basis to ensure the full engagement of the customer. Nevertheless, it is advisable to bear in mind that these technologies that appear to be almost flawless, suffer from numerous security and privacy issues to maintain and are also prone to a multitude of risks at the same time[7]. This part of the paper will deal with the newest technologies aimed at solving the mobile ML pipeline security issue and this will touch the problems that can arise in the future during the implementation of these solutions[4].

A. *Recent Advancements*

Homomorphic Encoding for Secure Computation
Homomorphic encryption enables computations on encrypted data without decryption, hence maintaining data privacy during the machine learning operation.
The Mechanism of Operation:

- Data is encrypted during processing, resulting in encrypted outputs.
- The client subsequently decrypts these results.
- The original data remains secure throughout[9].

B. *Applications:*

1. Healthcare

Security during the processing of sensitive patient data by applying ML models is made possible. The necessary inferences will be done using encrypted user preferences in personal recommendation systems without revealing raw data[6].
Homomorphic encryption is exceedingly resource-intensive and unsuitable for implementation on mobile devices with limited resources.

2. Blockchain for Secure Tracking of Model Updates

Blockchain technology provides a decentralized, unchangeable record for monitoring model revisions and facilitating safe communication.
It inhibits model tampering owing to the visible record of modifications[5].
It increases the federated learning systems' trust factor as it authenticates contributions from the participating devices.

Blockchain can be combined with federated learning for model update verification without compromising data privacy.
Blockchain implementation in mobile environments is difficult due to high storage and computational requirements.

3. Advanced Federated Learning Techniques

Federated learning is developing towards handling dynamic and heterogeneous mobile environments:
Dynamic Model Aggregation:
Hardware and user-data-dependent device-specific models.
Personalized Federated Learning:
Device-specific model learning while achieving global model accuracy.
Adaptive mobile learning, including virtual assistants, etc.
Collaborative ML in smart home[5].

C. *Future Challenges*

1. Balancing Personalization and User Privacy

Personalization always comes at the cost of a breach in user privacy. The secret to successfully implementing personalized modeling on a mobile is finding an optimal balance between personalization and user privacy.
Delivering context-aware and user-centric experiences requires collecting and analyzing user data.
Excessive data collection increases the risk of privacy violations and breaches[10].
Differential privacy to anonymize user data. Federated learning to enable personalized experiences without transferring raw data.

2. Handling Bias in ML Models

Bias in ML models may lead to unfair or discriminatory outcomes, especially in mobile applications targeting a broad spectrum of users[3].
Sources of Bias:
Training data skewed in such a way that it doesn't represent all the demographic groups. Algorithmic biases that tend to perpetuate stereotypes or inequities.
Exclusion, mistrust, and bad experience of users resulting from biased ML models.
Developing fairness-aware ML techniques in the construction of models and auditing deployed models for identified biases to remove[8].

3. Scalability of Security Structures

Challenges with scaling arise when implementing safe machine learning algorithms on millions of resource-constrained gadgets.
Cell phones generally have limited computational power, storage space and battery life. This makes it unsuited for highly complex operations such as sophisticated encryption or machine learning techniques.
The presence of more variation within both software and hardware complicates the deployment of protect guidelines.
System compressing techniques, including pruning and quantization, are employed to reduce computational overhead. Adaptable asset placement enhances efficiency across a variety of gadgets. Employing lightweight edge AI frameworks designed for particular hardware capacities[11].

D. *Mobile ML Pipelines Overview*

Mobile machine learning (ML) pipelines are important structures in most modern mobile applications that make intelligent services context-aware. The mobile pipelines consist of the following stages:
Data Collection Raw data is collected directly from the users via sensors, apps, and system logs. It is usually sensitive.

Data Preprocessing: Data cleaning and normalization. Preparation of data based on the demand of ML Model.
Model Training: Models are trained using datasets, which may include sensitive user information[10].
Model deployment: optimizes and then deploys trained models to devices or servers for direct, real-time inference.
Inference: Models generate predictions or actions based on user input and real-time data.
There are particular weaknesses attached to each of the development steps. Among the period of data gathering, unencrypted communication channels can pave the way to data leaks. Distorting pipelines allows for the embedding of poisoned data, leading to biased models that the rest of the process will examine. The privacy of model training can be interrupted: sensitive training data may be accidentally leaked. Deployment and inference stages hold threats of model theft, tampering, and adversarial attacks, by which the application could be incapacitated and users' trust could be breached[7].

Risks of Data Leakage and Adversarial Attacks

The increasing reliance on sensitive user data, together with the usage of cloud-based and direct-to-device ML processing, has led to a situation where mobile ML pipelines are becoming the main prey for security breaches. If there is insufficient encryption, insecure APIs, insiders who are threats, and data that is not properly protected, the data leakage will be imminent. For example, high-profile breaches such as those of the healthcare apps, where the violations of privacy have resulted in the exposure of private medical records[3].
The most important aggression, of course, is related to the creation of adversarial inputs by malicious actors, which leads to the deception of the ML models. In the area of mobiles, adversarial examples can be the cause of virtual assistants' misinterpretation of voice commands and recommendation systems suggesting a person's learning of inappropriate content. Research studies like the one by Goodfellow et al. (2015) show that the neural networks are one potential target for adversarial perturbations which can hardly be detected by humans[9].
One hears frequently about the grave risks associated with data breaches. For example, a famous fitness app in 2021 leaked the user location data accidentally while the AI apps such as the facial recognition system were under the radar for being infiltrated by adversarial attacks that exploit biases and vulnerabilities. These incidents underscore the pressing nature of forces requiring strong safety measures[6].

Safe Transportation and Archival Methods

Many strategies have been used to safeguard mobile devices machine learning workflows. These include:

- Guidelines for encryption RSA and AES are used to encrypt information while in transportation and at rest. These approaches have large computational overheads, making them difficult to apply on low-resource devices.
- Federated Training: Gadgets train machine learning models without sending raw data to servers. Federated training decreases data leakage but is subject to model poisoning and needs robust change communication.
- Model Compression: Quantizing and trimming model size for device security.
- These methods boost energy efficiency but reduce model accuracy if not done properly[8].

E. Existing research limitations

Despite this development, mobile ML pipeline security has significant gaps:

- Security vs. Performance Trade-offs: cryptography and federated learning engage computational and communication costs leading to a real-time mobile computing application which requires a lot of work to be able to use them[8].
- Adversarial Robustness: Mobile environments often see weak defenses against adversarial attacks, because shooting in real-time to create novel attacks is simpler for their authors.
- Bulletproof Model Deployment: Proving the correctness of the update process of the model during the production phase is almost always neglected, what to speak of the adversarial cases[3].

Prior research has laid the groundwork for safe mobile ML pipelines, but these challenges cannot be tackled completely by that. This paper is aimed at surmounting these obstacles by introducing innovative techniques and practical issues in realizing ML privacy-preserving and robust systems for smartphones[5].

VI. Conclusion

With mobile application developers turning to machine learning (ML) as their new weapon, the human experience is now digital and personalized with virtual assistance and healthcare solutions. Of course, the dynamic trends in technology have also presented many challenges, most of which are concerned with the security and privacy of mobile users who trade with such sensitive data. Bullet-proofing these weaknesses is critical to build and sustain the trust of the users and the proper functioning of mobile ML pipelines[7].

Construction of mobile ML technology elevates the security level by risks mitigation and innovation. Data encryption is a primary necessity because data confidentiality during storage and transmission is assured, and federated learning allows collaborative model training without raw user data being disclosed. To be more specific, adversarial training and model robustness techniques offer better security, keeping the application working properly in a fast-paced and often nemesis-filled digital world[5].

Despite these advantages, the quest for equilibrium between drive data security and allow high-speed data processing is a continuous problem. Ryan et al. reported that computational constraints on mobile devices, biases in ML models, and the complexities of real-time interactivity necessitate interdisciplinary research and innovation. Interdisciplinary collaborative support in areas such as cryptography, machine learning, hardware optimization, and legal compliance is a foundation to quicken the process of scale development and secure the needed ethical resolution[11].

Employing technologically advanced methodologies like homomorphic encryption and blockchain will significantly enhance the security of mobile ML pipelines. Furthermore, complying with new regulations and enforcing adherence but also promoting innovation is realized through the establishment of standardized guidelines and frameworks for secure ML deployment. Therefore, it becomes clear that with the continuous improvement of mobile ML, aesthetic quality and ethical aspects need to be set as priorities to ensure the sustainable growth and public trust in AI-powered applications will be.

Dealing with these challenges will lead to a future point where mobile ML pipelines not only add value for new users but are also fully compliant with all of the strict regulations concerning data and responsibility

References

[1] Z. W. Y. X. Z. S. &. K. G. Bu, "Automatic clipping: Differentially private deep learning made easier and stronger," *Advances in Neural Information Processing Systems,* vol. 36, 2024.

[2] J. M. Z. D. S. G. &. S. G. D. Chang, "Privacy-Preserving Machine Learning," *Simon and Schuster,* 2023.

[3] S. Z. L. G. &. D. H. El Mestari, "Preserving data privacy in machine learning systems," *Computers & Security,* no. 137, p. 103605, 2024.

[4] H. H. D. F. Q. K. M. K. L. M. &. C. K. K. R. Qin, "Cryptographic Primitives in Privacy-Preserving Machine Learning: A Survey.," *IEEE Transactions on Knowledge and Data Engineering.,* 2023.

[5] R. B. N. &. J. J. Xu, "Privacy-preserving machine learning: Methods, challenges and directions," *arXiv preprint arXiv:2108.04417,* 2021.

[6] J. L. N. &. R. B. Li, "Membership inference attacks and defenses in classification models," *In Proceedings of the Eleventh ACM Conference on Data and Application Security and Privacy,* pp. 5-16, 2021.

[7] E. O. B. G. N. &. C. R. Rodriguez, " A survey of deep learning techniques for cybersecurity in mobile networks," *IEEE Communications Surveys & Tutorials,* vol. 23, no. 3, pp. 1920-1955, 2021.

[8] R. U. A. H. F. R. W. Q. A. &. Q. J. Rasool, "Security and privacy of internet of medical things: A contemporary review in the age of surveillance, botnets, and adversarial ML," *Journal of Network and Computer Applications,* vol. 201, p. 103332, 2022.

[9] X. Z. Y. &. H. J. (. Yin, "A comprehensive survey of privacy-preserving federated learning: A taxonomy, review, and future directions," *ACM Computing Surveys (CSUR),* vol. 54, no. 6, pp. 1-36, 2021.

[10] F. S. J. &. R. C. Stodt, "Blockchain secured dynamic machine learning pipeline for manufacturing," *Applied Sciences,* vol. 13, no. 2, p. 782, 2023.

[11] Z. S. R. L. L. &. M. A. Sun, "Mind your weight (s): A large-scale study on insufficient machine learning model protection in mobile apps," *In 30th USENIX security symposium (USENIX security 21),* pp. 1955-1972, 2021.

[12] M. L. Z. A. &. Z. P. Zhang, "A secure and privacy-preserving word vector training scheme based on functional encryption with inner-product predicates," *Computer Standards & Interfaces,* vol. 86, p. 103734, 2023.