Practical Federated Recommendation Model Learning Using ORAM with Controlled Privacy

Mr. A. Karthick,
Assistant Professor, Computer Science and Engineering,
St. Joseph College of Engineering, Chennai-602117,
Tamil Nadu, Email Id – mnksjce@gmail.com

Ms. D. Nila,
Student, Computer Science and Engineering,
St. Joseph College of Engineering, Chennai-602117,
Tamil Nadu, Email Id – niladevaraj007@gmail.com

Ms. E. Saranya,
Student, Computer Science and Engineering,
St. Joseph College of Engineering, Chennai-602117,
Tamil Nadu, Email Id –
saranyasaranya878787@gmail.com

Abstract --Privacy preservation has become a major concern in modern recommendation systems due to centralized data collection and increasing risks of information leakage. Traditional recommendation frameworks collect user preferences, ratings, browsing history, and behavioural data on centralized servers, making them vulnerable to privacy breaches and inference attacks.
To address this issue, this project proposes a Practical Federated Recommendation Model Learning framework integrated with Oblivious RAM (ORAM) and Controlled Privacy mechanisms.
The system enables collaborative model training across distributed user devices without sharing raw data. Federated Learning allows each client device to train the recommendation model locally using personal interaction data. Only encrypted model updates are transmitted to the central server for aggregation.
To further strengthen privacy, ORAM is incorporated to hide memory access patterns during local training, preventing indirect leakage of sensitive information. Additionally, controlled privacy mechanisms such as gradient clipping and differential privacy noise are applied to ensure secure model updates.
The proposed system enhances data confidentiality, protects access patterns, reduces privacy risks, and maintains competitive recommendation accuracy, making it suitable for real-world deployment in secure recommendation platforms.

I. INTRODUCTION

Recommendation systems play a critical role in modern digital platforms such as e-commerce, streaming services, and social media.

These systems analyse user interaction data to generate personalized content suggestions. However, traditional centralized recommendation systems collect and store user data on central servers.

With increasing awareness of data protection and privacy regulations, there is a need for decentralized and privacy-preserving recommendation systems. Federated Learning (FL) offers a distributed learning approach where model training occurs locally on user devices. Instead of sending raw data, only model updates are shared with a central server.
Despite this improvement, federated learning alone does not protect against access pattern leakage and gradient-based privacy attacks. Therefore, this project integrates: Federated Learning, Oblivious RAM(ORAM)Controlled Privacy Mechanisms to design a secure and practical recommendation framework.

II. BACKGROUND AND MOTIVATION

Personalized Recommendation Systems

Recommendation systems are intelligent algorithms used in digital platforms to analyse user preferences and provide personalized suggestions. They enhance user experience by predicting relevant products, services, or content based on interaction history.

Centralized Learning Architecture

Traditional recommendation models operate using centralized architectures where user data is collected and stored on a central server. This approach enables large-scale data processing but increases the risk of privacy breaches and unauthorized data access.

Privacy and Security Risks

Centralized systems are vulnerable to cyber-attacks, data leaks, and internal misuse. Sensitive information such as browsing patterns, purchase history, and user behaviour may be exposed if the server is compromised.

Inference and Reconstruction Attacks

Advanced attacks such as model inversion and membership inference can extract sensitive information from trained models. Even without direct access to raw data, attackers may reconstruct user preferences from model outputs.

Federated Learning Concept

Federated Learning is a decentralized machine learning approach where model training is performed locally on user devices. Instead of sharing raw data, only model updates are transmitted to the server for aggregation.

Limitations of Basic Federated Learning

Although federated learning prevents raw data sharing, gradient updates may still leak information. Observing model parameters or updates can potentially reveal user-specific patterns.

Access Pattern Leakage Problem

During local training, repeated access to certain memory locations may indirectly reveal user interaction behaviour. Monitoring memory access patterns can compromise privacy even if data remains encrypted.

Oblivious RAM (ORAM) Technology

ORAM is a privacy-preserving technique that hides memory access patterns by randomizing data access and introducing dummy operations. It ensures that real and fake accesses are indistinguishable to observers.

Controlled Privacy Mechanisms

Techniques such as gradient clipping and differential privacy add controlled noise to model updates. This reduces the possibility of reconstructing individual user data from shared parameters.

Project Motivation

The motivation of this project is to design a secure and practical federated recommendation framework that integrates ORAM and controlled privacy techniques. The goal is to protect raw data, prevent gradient leakage, hide access patterns, and maintain high recommendation accuracy in real-world applications.

III. SYSTEM ARCHITECTURE

The proposed architecture for **Practical Federated Recommendation Model Learning Using ORAM with Controlled Privacy** is designed to ensure secure, privacy-preserving, and scalable recommendation generation. The system follows a decentralized federated learning framework enhanced with ORAM-based access protection and controlled privacy mechanisms.

A. ARCHITECTURE COMPONENTS

1. **Client Devices**
   Client devices represent end-user systems such as smartphones, laptops, or edge devices. Each client stores its own private interaction data (ratings, clicks, browsing history). Raw data never leaves the device, ensuring primary data privacy.

2. **Local Training Module**
   The local training module is responsible for updating the recommendation model using the client's private dataset. The global model received from the server is trained locally using gradient-based optimization techniques. Only model updates (not raw data) are prepared for transmission.

3. **ORAM Protection Layer**
   The ORAM (Oblivious RAM) protection layer hides memory access patterns during local computation. It prevents attackers from inferring sensitive user information based on data access frequency or patterns. ORAM introduces dummy accesses and randomized memory operations to ensure indistinguishability.

4. **Secure Communication Channel**
   A secure communication channel encrypts model updates before transmission to the central server. Encryption techniques such as TLS/SSL protect against eavesdropping, replay attacks, and tampering.

5. **Secure Aggregation Server**
   The aggregation server collects encrypted updates from multiple clients. Using secure aggregation protocols (e.g., Federated Averaging), it combines model updates without accessing individual client parameters.

6. **Global Model Distribution**
   After aggregation, the updated global model is redistributed to all participating clients. This iterative process continues for multiple training rounds until convergence is achieved.

B. WORKFLOW OF THE SYSTEM

The overall system workflow is described below:

User Interaction → Local Training → ORAM Protection → Encrypted Update → Secure Aggregation → Global Model Update → Recommendation Delivery

Step-by-Step Explanation:

1. **User Interaction**
   Users interact with the platform (viewing, rating, purchasing items). These interactions are stored locally.

2. **Local Training**
   The client trains the global recommendation model using its private data and generates gradient updates.

3. **ORAM Protection**
   During training, ORAM ensures memory access patterns are concealed to prevent side-channel leakage.

4. **Encrypted Update**
   The trained model updates are encrypted before transmission.

5. **Secure Aggregation**
   The central server securely aggregates updates from multiple clients without accessing individual data.

6. **Global Model Update**
   The server updates the global recommendation model using aggregated parameters.

7. **Recommendation Delivery**
   The updated global model is sent back to clients to generate personalized recommendations.

IV. MODULE DESCRIPTION

The proposed Practical Federated Recommendation Model Learning system with ORAM and Controlled Privacy is structured into multiple interconnected modules. Each module is designed to ensure decentralized learning, secure communication, and strong privacy protection while maintaining high recommendation performance.

The first module is the **Local Data Collection Module**. This module operates entirely on the client side and is responsible for collecting user interaction data such as ratings, clicks, browsing history, search behaviour, and purchase records. Unlike traditional centralized systems, this data is stored locally on the user's device and is never transmitted to the server. The module also performs basic preprocessing tasks including data normalization, filtering incomplete entries, and structuring user-item interaction matrices. By keeping all raw data locally, the system ensures primary privacy protection at the data source level.

The second module is the **Local Model Training Module**. In this stage, each client device receives the global recommendation model from the server and trains it using its local dataset. The model may use collaborative filtering, neural embeddings, or deep recommendation architectures to learn user preference patterns. During training, the module computes loss functions and generates gradient updates based on user-item interactions. This decentralized training mechanism allows collaborative improvement of the recommendation model without exposing personal user information.

The third module is the **ORAM-Based Privacy Protection Module**. This module enhances privacy by hiding memory access patterns during local training. Traditional memory access operations may reveal sensitive behavioural information if monitored by an attacker. The ORAM mechanism randomizes memory block access, introduces dummy operations, and reshuffles stored parameters to prevent correlation between access patterns and actual user interactions. By ensuring that real and dummy accesses are indistinguishable, this module protects against side-channel and access pattern inference attacks.

The fourth module is the **Controlled Privacy and Gradient Protection Module**. After local training, gradient clipping is applied to limit the sensitivity of model updates. Differential privacy noise is then added to gradients to prevent reconstruction of individual user data from transmitted updates. This module carefully balances privacy and model accuracy by controlling the noise level through privacy budgets. It ensures that even if model updates are analysed, user-specific contributions remain unidentifiable.

The fifth module is the **Secure Communication Module**. This module encrypts the locally trained model updates before transmission to the server. Secure communication protocols such as SSL/TLS are used to prevent interception during data transfer. The encrypted updates guarantee that sensitive parameter information is protected while moving between client devices and the aggregation server.

The sixth module is the **Secure Aggregation Module**. At the server side, encrypted updates from multiple clients are aggregated using a federated averaging strategy. The secure aggregation protocol ensures that individual updates cannot be accessed or reconstructed by the server.
Only the aggregated global model parameters are computed, preserving collaborative learning while maintaining privacy guarantees.

The final module is the **Global Recommendation and Redistribution Module**. After aggregation, the updated global model is redistributed to participating clients. This model is then used to generate personalized recommendations locally. The iterative training process continues across multiple communication rounds until convergence is achieved.

Together, these modules form a comprehensive, multi-layered privacy-preserving recommendation framework that protects raw data, hides access patterns, secures gradient updates, and ensures scalable federated learning deployment.

.

V. IMPLEMENTATION METHODOLOGY

The implementation of the proposed Practical Federated Recommendation Model Learning framework with ORAM and Controlled Privacy is carried out through a structured multi-stage process that integrates decentralized learning, privacy-preserving computation, and secure aggregation mechanisms. The methodology is designed to ensure that user data remains locally protected while collaborative model improvement is achieved efficiently and securely.

The implementation begins with the initialization of a global recommendation model at the central server. This model may be based on matrix factorization, neural collaborative filtering, or embedding-based architectures capable of learning user–item interaction patterns. The server distributes the initialized global model parameters to all participating client devices. Each client device represents an individual user

environment containing local interaction data such as ratings, browsing behaviour, and preference history. Importantly, this raw data never leaves the client device.

Once the global model is received, each client performs local model training using its private dataset. During this phase, the system computes gradients or parameter updates based on user-item interaction loss functions. To enhance privacy protection during computation, gradient clipping is applied to limit the magnitude of updates, preventing excessive information exposure. Additionally, differential privacy mechanisms introduce controlled random noise into the gradients, ensuring that individual user contributions cannot be reconstructed from model updates.

To further strengthen privacy, an Oblivious RAM (ORAM) layer is integrated into the local training process. ORAM ensures that memory access patterns during model training remain hidden. This is achieved by randomizing memory block access, introducing dummy operations, and reshuffling stored parameters. As a result, even if an adversary observes system-level access behaviour, they cannot infer which user-item interactions are being processed. This prevents indirect leakage of sensitive information through access pattern analysis.

After local training and ORAM-protected computation, the updated model parameters are encrypted before transmission. Secure communication protocols such as SSL/TLS ensure safe data transfer between clients and the central server. The encrypted updates are then sent to the server for aggregation.
At the server side, a secure aggregation mechanism combines the encrypted updates from multiple clients without revealing any individual contribution. Aggregation typically follows a federated averaging strategy, where model updates are weighted and averaged to produce a new global model. Since updates remain encrypted during aggregation, the server cannot access user-specific information.

The updated global model is then redistributed to participating clients for the next training round. This iterative process continues for multiple communication rounds until the model converges to an optimal performance level. Throughout the process, privacy preservation is maintained at three levels: raw data protection (no data sharing), gradient protection (differential privacy), and access pattern protection (ORAM).

Finally, system performance is evaluated using recommendation accuracy metrics such as precision, recall, F1-score, and root mean square error (RMSE). Privacy effectiveness is analyzed by examining resistance to gradient leakage and inference attacks. Computational overhead introduced by ORAM and encryption mechanisms is also measured to ensure practical feasibility.

This structured implementation methodology ensures that the proposed system achieves secure, scalable, and privacy-preserving recommendation model learning suitable for real-world deployment in sensitive digital platforms.

VI. RECENT ADVANCEMENTS IN FEDERATED LEARNING AND PRIVACY

Recent advancements in federated learning and privacy-preserving machine learning have significantly strengthened the development of secure recommendation systems. Modern federated optimization algorithms such as Fed-Prox, Scaffold, and momentum-based federated averaging have improved convergence speed and training stability, especially in environments where client data is highly heterogeneous. These techniques reduce model drift and communication overhead, making large-scale federated recommendation systems more practical for real-world deployment. Such improvements directly enhance the efficiency of decentralized recommendation training without compromising performance.

Secure aggregation protocols have also evolved considerably in recent years. Advanced cryptographic methods such as secure multi-party computation, secret sharing schemes, and homomorphic encryption now enable encrypted model updates to be aggregated without revealing individual client contributions. These techniques ensure that even the central server cannot inspect or reconstruct user-specific gradients. As a result, federated recommendation systems can provide stronger protection against gradient inversion and inference attacks, which were previously major privacy concerns in collaborative learning frameworks.

Another major advancement is the integration of differential privacy into federated learning environments. Modern privacy-preserving systems implement adaptive noise injection, gradient clipping strategies, and privacy budget management to control information leakage.
These mechanisms mathematically guarantee that individual user contributions cannot be identified from model updates. In recommendation systems, this prevents attackers from reconstructing sensitive user preferences such as browsing habits, purchase history, or content interests. The balance between privacy protection and recommendation accuracy has improved significantly due to optimized noise calibration techniques.

Recent developments in Oblivious RAM (ORAM) have also contributed to practical deployment feasibility. Earlier ORAM implementations were computationally expensive and unsuitable for edge devices. However, optimized approaches such as Path ORAM and lightweight recursive ORAM have reduced memory overhead and access latency. These improvements allow ORAM to be integrated into federated client devices, effectively hiding memory access patterns during local model training. This is particularly important in recommendation systems, where frequent access to certain items or user embeddings could otherwise reveal sensitive behavioural trends.

Edge computing advancements further support federated recommendation frameworks.

With the rise of powerful mobile processors and optimized neural architectures, local model training can now be performed efficiently on smartphones and IoT devices. Techniques such as model compression, quantization, and knowledge distillation reduce computational requirements while maintaining accuracy. This makes decentralized recommendation learning scalable and energy-efficient.

In addition, modern recommendation models increasingly utilize transformer-based architectures and self-attention mechanisms to capture sequential and contextual user behavior. Integrating these advanced models into federated frameworks enhances personalization quality while maintaining privacy. Furthermore, recent research focuses on detecting and mitigating adversarial attacks in federated environments, including membership inference and model poisoning attacks, thereby improving system robustness.

Overall, the combination of advanced federated optimization, secure aggregation, differential privacy, lightweight ORAM techniques, and edge-based deployment has transformed privacy-preserving recommendation systems from theoretical concepts into practical, scalable solutions.

VII.CHALLENGES

The implementation of the proposed Practical Federated Recommendation Model Learning Using ORAM with Controlled Privacy system presents several technical and practical challenges. One of the primary challenges is maintaining a balance between privacy and accuracy, as strong privacy mechanisms such as differential privacy and ORAM may introduce computational overhead or controlled noise that can affect recommendation performance.

The integration of ORAM significantly increases computational complexity due to dummy memory accesses and reshuffling operations, which can be demanding for resource-constrained client devices like smartphones. Additionally, federated learning requires frequent communication between clients and the central server, leading to higher bandwidth consumption and communication delays, especially in large-scale deployments. Secure aggregation protocols must be carefully designed to prevent the server from accessing individual client updates while still ensuring correct model averaging, which adds implementation complexity.

The system must also handle device heterogeneity, as clients may differ in processing power, storage, and network connectivity. Ensuring robustness against advanced attacks such as gradient inversion, membership inference, and side-channel attacks remains a continuous security challenge. Moreover, non-IID data

distribution across clients can slow down model convergence and affect stability. Energy consumption on edge devices and the overall scalability of the system further complicate real-world deployment. Addressing these challenges is essential to ensure that the system remains secure, efficient, and practically deployable.

VIII. CONCLUSION

In conclusion, the proposed Practical Federated Recommendation Model Learning Using ORAM with Controlled Privacy framework provides a secure and efficient approach to building privacy-preserving recommendation systems. By combining federated learning with ORAM-based access pattern protection, controlled privacy mechanisms, and secure aggregation protocols, the system ensures that sensitive user data remains protected at multiple levels without compromising recommendation quality. The architecture prevents raw data sharing, reduces the risk of gradient leakage, and safeguards memory access patterns, making it suitable for real-world deployment. Overall, the proposed system achieves a balanced trade-off between privacy, security, and performance, offering a practical solution for next-generation personalized recommendation platforms.

IX. REFERENCE

1. McMahan, B., Moore, E., Ramage, D., Hampson, S., & Aguera y Arcas, B. (2017). *Communication-Efficient Learning of Deep Networks from Decentralized Data*. Proceedings of the 20th International Conference on Artificial Intelligence and Statistics (AISTATS).
2. Goldreich, O., & Ostrovsky, R. (1996). *Software Protection and Simulation on Oblivious RAMs*. Journal of the ACM, 43(3), 431–473.
3. Dwork, C. (2006). *Differential Privacy*. Proceedings of the 33rd International Colloquium on Automata, Languages and Programming (ICALP).
4. Bonawitz, K., et al. (2017). *Practical Secure Aggregation for Privacy-Preserving Machine Learning*. Proceedings of the ACM Conference on Computer and Communications Security (CCS).
5. Kairouz, P., et al. (2021). *Advances and Open Problems in Federated Learning*. Foundations and Trends® in Machine Learning, 14(1–2), 1–210.
6. Wang, T., et al. (2024). *Privacy-Preserving Cross-Domain Federated Recommendation*. IEEE Transactions on Knowledge and Data Engineering.
7. Li, T., Sahu, A. K., Talwalkar, A., & Smith, V. (2020). *Federated Learning: Challenges, Methods, and Future Directions*. IEEE Signal Processing Magazine, 37(3), 50–60