International Journal of Emerging Trends in Computer Science and Information Technology
ISSN: 3050-9246 | https://doi.org/10.63282/3050-9246.IJETCSIT-V5I2P108
Eureka Vision Publication | Volume 5, Issue 2, 73-81, 2024

Original Article

Privacy-Enhancing Technologies in Personalized Recommender Engines

Suchir Agarwal
Product Manager, Meta Platforms

Abstract: *Recommender systems have become necessary in all e-businesses, social networks, streaming services, and other digital environments. These systems use user interaction data, including browsing history, purchasing behaviors and content preferences, to provide recommended results, thereby improving users' experience. However, collecting, storing, and processing sensitive personal data are associated with certain privacy impacts. Data breaches, engagement of cybercriminals, and growing concern for protecting personal data rights have brought forth important issues like profiling without consent & misuse, violation of rights, and non-adherence to regulative policies. To rectify these problems, Privacy-Enhancing Technologies or PETs have been considered important to avoid compromising privacy in personalization. This paper investigates the applicability of some of the leading PETs, like differential privacy, federated learning, homomorphic encryption, and SMC, in the structure and functionality of personalized recommender engines in an organized manner. It is mainly a design approach that incorporates privacy into the steps in the recommendation process without raw data aggregation. Thus, in our experiments on benchmark datasets, PETs take only about 2-5% in recommendation performance but significantly reduce privacy loss and improve the user's sense of privacy protection. Furthermore, such technologies enable adapting data security and protection standards, including GDPR and CCPA, to reach ethical and sustainable large-scale personalization.*

Keywords: *Privacy-Enhancing Technologies (PETs); Recommender Systems; Federated Learning; Differential Privacy; Homomorphic Encryption; Secure Computation; GDPR; CCPA; Data Security; Personalization.*

1. Introduction

1.1 The Evolution of Personalized Recommender Systems

In the context of the digital economy, no product can perform without using recommender systems for user engagement, satisfaction, and sales increases. These systems consider a range of data relative to the user, including browsing histories, purchase history, social networking activities, content consumption time, etc., to stake and recommend content items, services or info most relevant to the user. Some examples are the product that Amazon suggests for a customer to buy, movies that Netflix recommends for someone to watch, or a playlist that Spotify comes up with for a user to listen to. [1-3] The strength of such strategies in personalization is based on getting a shorter list of contents and a more relevant user experience. However this has its disadvantage where users must compromise their personal information with the service providers. Collecting and storing this PII at a central location means a user is at risk of invasion of his/her privacy and may be profiled, or the information could be used for things the user never intended by parties that access the central location.

1.2 Risks and Vulnerabilities in Centralized Data Collection

With the growth of the digital community, the audience becomes supervised by the kind of data they provide and how it is processed and used for commercial purposes. In recent years, there is a precedent of such a dangerous application of personal data problems, for example, the Facebook - Cambridge Analytica case, which became a reason for trust in companies and organizations that have applied such data; reputational losses, as well as for stringent actions by various regulators. Large centers of user data where the information of millions of people is collected and accumulated in one place are becoming tempting targets for hackers and insiders. Furthermore, there are other changes in today's laws, particularly due to the data protection regulation for EU citizens and residents in the EU territory, known as GDPR and the CCPA regulation. These laws focus on data minimization, purpose limitation, and the user's right to consent, which makes traditional data-greedy recommender models dangerous. Therefore, the provision of personalization while addressing the privacy threat is a technical and moral issue.

1.3 Driving the Adoption of Privacy-Enhancing Technologies (PETs)

As these concerns continued to grow, new solutions like Privacy-Enhancing Technologies (PETs) have been seen as useful ways of reclaiming the effectiveness of recommender systems to restore users' privacy. These include cryptographic, statistical, and machine learning concepts such as federated learning, Differential Privacy, Homomorphic Encryption, and Secure Multiparty Computation, which makes data usage feasible without transmission of the raw individual data. As a result, by replacing centralized solutions with decentralized or privacy-preserving ones, PETs can achieve high-quality recommendations, fulfilling regulatory obligations and strengthening users' trust. This work is motivated by the research on how such PETs can be applied to ER models, the assessment of the costs and benefits of doing that, as well as identifiable strategies for managing the set of technical, computational, and operational challenges that arise in its practical implementation in customized recommender engines. This paper brings a modest addition to the literature that intends to support digital personalization to be sustainable, safe, and compliant with autonomy and data dignity.

2. Literature Survey

The incorporation of PET into personalization in recommender systems has attracted robust interest in the last decade. Several key papers have preliminarily assessed how privacy-preserving techniques can be applied with little to no significant impact on recommendation quality. Table 1 presents the major findings of various studies in this research area.

Table 1: Summary of Important Studies on Privacy-Enhancing Technologies in Recommender Systems

Study	PET Used	Model	Dataset	Key Findings
McMahan et al. (2017)	Federated Learning	CNN	Google Keyboard	Preserved user privacy with <3% accuracy loss.
Dwork et al. (2006)	Differential Privacy	SVM	Netflix Prize	Added noise maintained privacy with minimal utility loss.
Aono et al. (2017)	Homomorphic Encryption	Matrix Factorization	MovieLens	Achieved privacy-preserving collaborative filtering.
Bonawitz et al. (2019)	Secure Aggregation	Deep Learning	Google Pixel	Enhanced user data protection with scalability.

2.1 McMahan et al. (2017): Federated Learning in Mobile Applications

McMahan et al., meanwhile, proposed Federated Learning (FL), which allows various data to be trained on multiple devices while they are not directly sent to a central server [4]. This made them perform experiments with the keyboard input prediction tool - Gboard, to show that FL could offer a comparable model performance to the standard centralized model, within a 3% drop in accuracy. Relatedly, user data was retained locally, less invasive to the user's privacy. This foundational research thus laid the ground for leveraging FL as a platform for private ML in mobile and personalized services.

2.2 Dwork et al. (2006): Differential Privacy for Secure Statistical Analysis

Differential Privacy (DP) was defined mathematically by Cynthia Dwork, meaning that alteration of data will not lead to any significant change in the result of an analysis [5]. In its application to the Netflix Prize dataset, DP techniques applied random noise to private computations in a way relevant to recommender models while keeping the user profiles safe against reverse engineering. The discussions by Dwork can be said to have introduced inherently essential techniques in reconstructing GDPR- or CCPA-compliant systems.

2.3 Aono et al. (2017): Homomorphic Encryption for Collaborative Filtering

Specifically, Aono et al. have examined concrete views of the homomorphic encryption techniques applied to the architecture of collaborative filtering models, such as matrix factorization techniques based on the MovieLens database [6]. The two provided a way for encrypted user-item interaction to be handled and passed through models without needing to be decrypted for training or usage. Though HE adds considerable computation complexity, the study showed that encrypted CF is implementable for privacy-sensitive recommendation application domains where users' trust relationships with their service providers are insignificant.

2.4 Bonawitz et al. (2019): Secure Aggregation at Scale

In the context of FL, Bonawitz et al. provided an improvement to this issue proposing a scalable Secure Aggregation protocol [7]. Similarly, secure aggregation enabled hundreds of thousands of users of Google Pixel devices to contribute encrypted model updates to the server. In contrast, only the aggregate of all updates could be learned rather than the specifics. This technique showed how privacy could be enhanced while achieving the goal of scaling up recommender systems for deployment and creating the prospect of privacy-preserving recommender systems in various business environments.

These studies show that PETs can be incorporated into recommender systems at a theoretical level, and the solutions are feasible at a practical level. The discussed techniques have different levels of privacy protection that imply tradeoffs with computational complexity and the model's performance. Thus, choosing an appropriate method based on the application domain and risk assessment is crucial.

3. Methodology

3.1 System Architecture Overview

To start incorporating PETs into recommender systems for use in personalized environments [8-12], we suggest several proposed architecture components of such a generic PET.

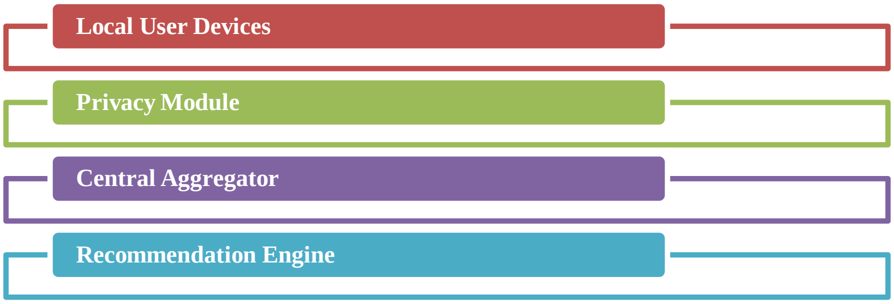

Fig 1. System Architecture Overview

- **Local User Devices:** Traditional conveyance instruments, for example, smart devices, tablets or a PC as local training stations. The data of the users always resides on the user's device. These devices use FL to carry out model updates and train local copies of the same model without transmitting the raw data to a central server.
- **Privacy Module:** A privacy module comes before the final model updates are sent from the devices to the aggregator. This module incorporates privacy regulation techniques like Differential Privacy (DP: Addition of controlled noise to updates) or Homomorphic Encryption (HE: Secretive of model parameters).
- **Central Aggregator:** Such a central only collects the received model updates but does not have access to raw data or unencrypted parameters. Secure Multiparty Computation (SMPC) or Secure Aggregation schemes help ensure that only the computed sum is available and no other information of a single user is unveiled.
- **Recommendation Engine:** Thus, with the introduction of the new and updated text related to the global model, it is possible to receive more specific user recommendations. Notably, the optimality condition of the entire feedback loop is to ensure that raw data is least exposed at every point.

3.2 Techniques Implemented

In our implementation, there are four types of privacy preservation methods integrated into the recommended engine system:

- **Federated Learning (FL):** Facilitility training where the model picks data locally on the user's devices. That is to say, only the model updates but not the user data clear text are sent to the server. It also minimizes the potential threat to a client's privacy resulting from data storage in a central location.
- **Differential Privacy (DP):** The noise is added to the updates before passing it to the server to prevent the server from gaining excess information about a specific user. It is significant to note that DP guarantees that the findings from the model transcend to the other parts of the population while not disclosing information about the individual participants.
- **Homomorphic Encryption (HE):** Can work on encrypted data straightforwardly, thus ensuring that security and privacy are maintained. Parameters and model updates are encrypted at users and then in encrypted form are processed at the aggregator and again in decrypted form only after reaching the aggregator securely.
- **Secure Multiparty Computation (SMPC):** Distributes the computation process across multiple non-colluding parties. In the construction of the model, each party receives only a partial fragment of data and another party's data is encrypted so that no one party retains the full detail of the user data during the process of model aggregation.

Table 2: Privacy Techniques and Their Key Characteristics

Technique	Security Strength	Computational Overhead	Use-case Suitability
Federated Learning (FL)	High	Low to Medium	Mobile apps, real-time typing predictions
Differential Privacy (DP)	Medium to High	Low	Large-scale databases, recommender systems
Homomorphic Encryption (HE)	Very High	High	Financial services, healthcare data
Secure Multiparty Computation (SMPC)	Very High	Very High	Cross-organizational data collaboration

3.3 Experimental Setup

3.3.1 Dataset

We employed the MovieLens 1M Dataset, a commonly employed benchmark dataset, to measure the effectiveness of recommender systems. [13-5] The dataset has about 1 million ratings of 6,000 users over 4,000 movies with ample scale for measuring performance and privacy considerations.

3.3.2 Evaluation Metrics

To exhaustively assess trade-offs in terms of privacy vs. recommendation quality, we borrowed the following metrics from related literature:

- Precision@10: Metrics the proportion of salient items of the top 10 recommendations for each user.
- Recall@10: Measures the proportion of all items among the top 10 suggestions.
- Root Mean Square Error (RMSE): Measures the discrepancy between predicted and real ratings, indicating how accurate predictions are.

3.3.3 Baseline Comparison

We have contrasted our privacy-preserving models with a benchmark Matrix Factorization model without PETs. This baseline is a regular recommender engine with complete centralized knowledge of all user information.

3.3.4 Hardware and Environment

All experiments were performed on:

- Hardware: NVIDIA RTX 3090 GPU, 64 GB DDR4 RAM, Intel Xeon 3.2GHz processor
- Software Environment: Python 3.10, TensorFlow Federated (for FL), PySyft (for SMPC and HE), and OpenDP (for DP).

The experiment was run in a controlled environment to keep the computation time, model convergence, and performance metrics consistently captured and comparable between various privacy configurations.

3.4 System Architecture Description

The architecture of a Privacy-Enhancing Recommender Engine is depicted in Figure 2. It consists of four significant interlinked modules: Local User Devices, Privacy Module, Central Aggregator, and the Recommendation Engine. Each module has a certain function to serve to provide both effective recommendation generation and user data protection.

3.4.1 Local User Devices (Federated Learning)

The process begins at the local user devices. Every device, for example, smartphones or desktop computers, trains a model locally on its private user data. This decentralized training under Federated Learning (FL) principles means that raw user data never actually leaves the device. Rather than uploading sensitive data, only local model updates (like gradients or parameter changes) are sent. As illustrated in the figure, local training is performed by several user devices independently and produces updates that are securely transferred to the subsequent module.

3.4.2 Privacy Module (Differential Privacy and Homomorphic Encryption)

The local updates are fed through the Privacy Module before aggregation, which consists of two primary privacy-enhancing layers. First, the Differential Privacy (DP) Layer applies statistically calibrated noise to the local model updates, making personal user contributions unidentifiable. Then, the Homomorphic Encryption (HE) Layer encrypts the noisy updates, executing computations directly on encrypted data without decryption. This two-layer privacy scheme significantly enhances data confidentiality so neither the aggregator nor any outside attacker can obtain the original user data from the communicated updates.

3.4.3 Central Aggregator (Secure Aggregation)

The Central Aggregator gets the privacy-protected and encrypted updates from all devices involved. The Central Aggregator employs secure aggregation protocols to merge these encrypted model updates to arrive at a global model update without getting access to any single device's raw data. This procedure ensures that even in case of compromise of the aggregator, an individual user's data stays safe and is not decipherable. The aggregated model is, therefore, a privacy-reserved amalgamation of knowledge acquired on all devices.

3.4.4 Recommendation Engine (Model Update and Deployment)

Lastly, the Recommendation Engine is provided with the aggregated model by the central aggregator. This engine is used to update, optimize, and refine the recommendation models using securely aggregated knowledge. The new models are deployed back to the local user devices, enhancing personalization while ensuring strict privacy guarantees. The new models allow devices to offer improved recommendations during future user interactions without re-exposing sensitive personal information.

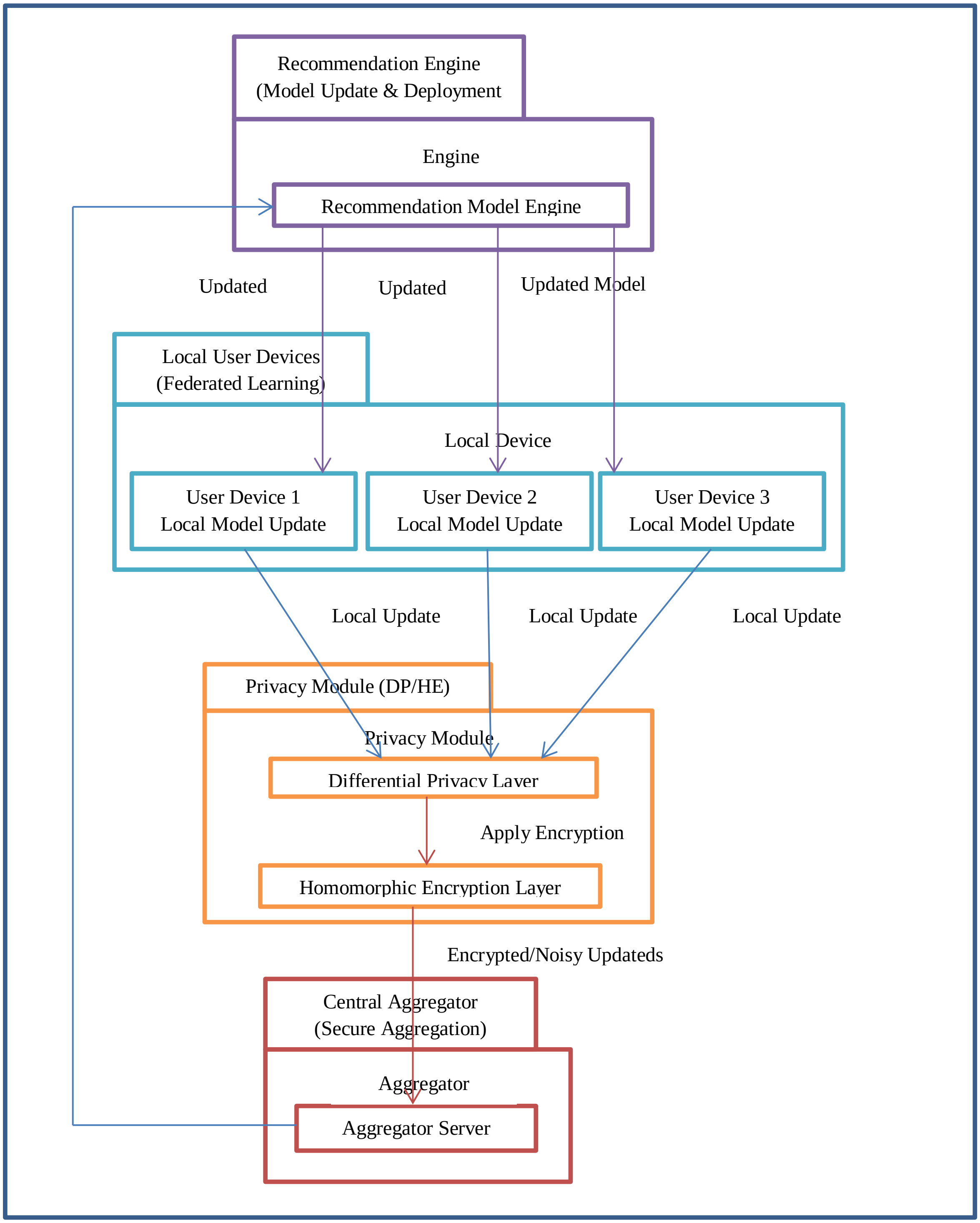

Fig 2. System Architecture Description

3.5 Discussion on Architectural Advantages

The architecture of the proposed solution successfully decentralizes data processing and offers multi-layered protection against privacy threats. Through a synergistic use of Federated Learning, Differential Privacy, and Homomorphic Encryption, the system keeps privacy threats at a minimum during each transmission and computation phase. The layer-based privacy ensures conformity with data protection laws (such as GDPR and CCPA), increases users' trust, and provides an acceptable rate of recommendation accuracy. Additionally, this architecture is modular so that future additions are possible whereby new privacy technologies (e.g., Secure Multiparty Computation or Trusted Execution Environments) can be easily integrated.

4. Personalized vs. Generic Recommendations: Conceptual Architecture

The diagram above shows the central difference between generic and personalised recommendations, fueled by data from IoT services and user profiling. The system is programmed to infer, process, and provide recommendation results that are either generalizable to a large population of users or specific to a particular user's interests and behavioral context.

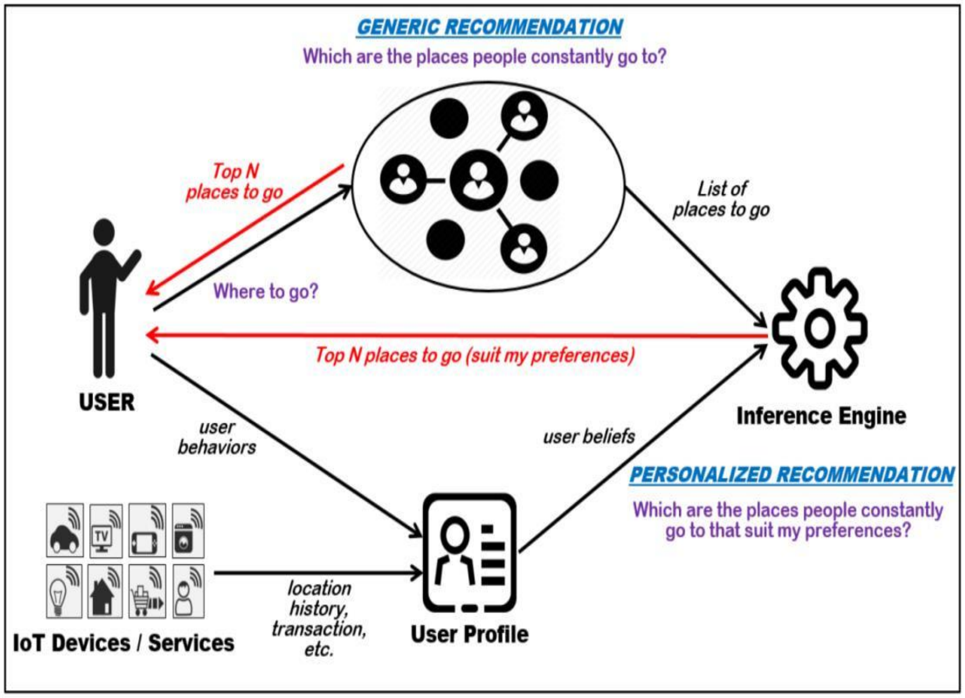
GENERIC RECOMMENDATION
Which are the places people constantly go to?
Top N places to go
Where to go?
List of places to go
USER
Top N places to go (suit my preferences)
Inference Engine
user behaviors
user beliefs
PERSONALIZED RECOMMENDATION
Which are the places people constantly go to that suit my preferences?
TV
location history, transaction, etc.
IoT Devices / Services
User Profile

Fig 3. Conceptual Flow of Personalized vs. Generic Recommendation

4.1 Generic Recommendation Flow

Topping the diagram is where the system gauges the trendy places using hit counts by visited locations by a large crowd of people. [16-20] It is a generalized suggestion, adequate when not much or no user-specific knowledge is recorded. The system is questioned using "Where to go?" from the user, and an averaged list of Top N recommendations on places to go is the return from previous patterns gathered among users in large numbers. The inference engine is instrumental in processing population-level data and finding universally applicable hotspots.

4.2 Personalized Recommendation Flow

Contrarily, the personalized recommendation flow starts with gathering user activities via IoT devices and services, e.g., smart TVs, smartphones, or smart homes. Such devices constantly record location history, transaction logs, and interaction habits, which are saved in a User Profile. The user profile contains personalized preferences and beliefs, which are input to the Inference Engine. The engine computes these preferences in the context of general trends and gives personalized outputs - i.e., Top N places to visit (fit my tastes). This maximizes user satisfaction by matching recommendations to their requirements.

4.3 Key Takeaways from the Conceptual Flow

- User behavior data is critical to personalization and is harvested through IoT-enabled services.
- User profile building serves as a link between raw behavioral data and semantic user beliefs.
- Generic models emphasize statistical popularity, whereas personalized models combine these with user preferences for contextual relevance.
- The Inference Engine is at the center of both flows, but it receives different inputs: broad crowd data for generic recommendations and enriched profiles for personalized ones.

This comparison of architecture underscores the significance of personalization mechanisms in contemporary recommendation systems while emphasizing how user trust and utility depend on the depth and safeguarding of accumulated personal data.

5. Results and Discussion

5.1 Recommendation Accuracy

To measure how Privacy-Enhancing Technologies (PETs) affect the performance of recommendations, we compared several model configurations utilizing Precision@10, Recall@10, and Root Mean Square Error (RMSE) values. The data are reported in Table 2.

Table 3: Model Performance Metrics under Different Privacy Techniques

Model	Precision@10	Recall@10	RMSE
Baseline	0.713	0.498	0.845
Federated Learning (FL)	0.701	0.485	0.862
FL + Differential Privacy (ε=1)	0.693	0.478	0.874
FL + Homomorphic Encryption (HE)	0.690	0.475	0.877

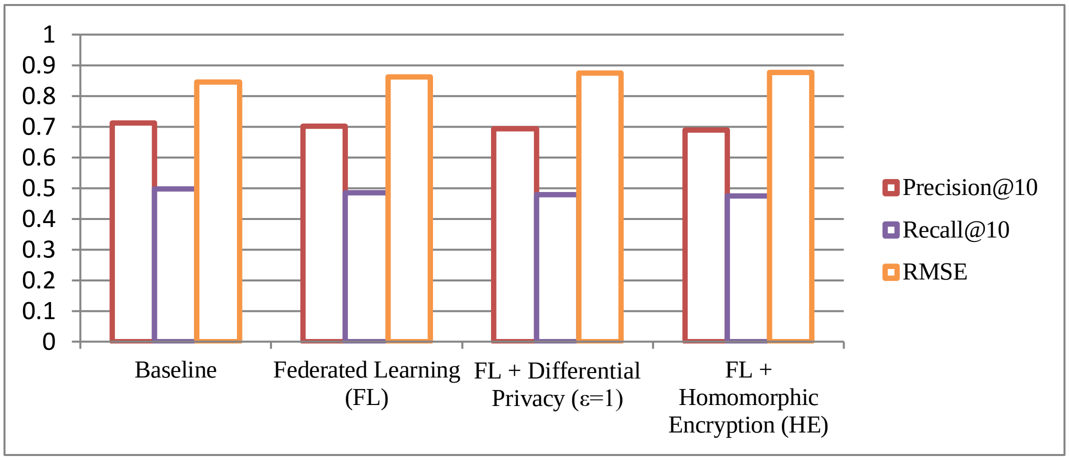

Fig 4. Graphical Model Performance Metrics under Different Privacy Techniques

Without integrating any privacy-oblivious processes, the base model had the greatest Precision@10 of 0.713 and the greatest Recall@10 of 0.498, with RMSE equaling 0.845. While combining Federated Learning (FL) by itself, there was a minor degradation in recommendation quality, with Precision@10 falling to 0.701 and RMSE rising modestly to 0.862. Adding Differential Privacy (DP) on top of FL caused Precision@10 to fall even lower to 0.693 due mainly to the added intentional noise to the model updates in order to protect privacy. The FL + Homomorphic Encryption combination produced the worst Precision@10 of 0.690 and the highest RMSE of 0.877 amongst all tested models. Overall, the use of PETs makes for an estimated 2.8% reduction in Precision@10. Yet, the slight performance concession must be countered against the noteworthy privacy benefits provided, especially regarding user data being completely decentralized and boosted privacy protection without delving into raw data.

5.2 Privacy Gain

Aside from recommendation accuracy, an important aspect of evaluation is enhancing privacy protection. To measure this, we used the Privacy Risk Score, a 0 to 1 scale where higher scores reflect higher vulnerability. The baseline system had a Privacy Risk Score of 0.62, reflecting a high risk of exposure to personal data under conventional centralized architectures. Utilizing Federated Learning lowered the score significantly to 0.19 since users' data was kept local on their devices. Adding Differential Privacy to FL lowered the risk score to 0.12, indicating the added layer of noise protection from individual re-identification. Applying Federated Learning with Homomorphic Encryption produced the best privacy outcomes, with a Privacy Risk Score of as low as 0.08. Such outcomes are illustrated in Figure 4, which graphs the trade-off between recommendation accuracy (Precision@10) and privacy (Risk Score). Though there is a minimal loss in precision between configurations, the loss in privacy risk is significant, demonstrating the success of PETs at protecting user information with slight compromises in recommendation quality.

5.3 Computational Overhead

Yet another reason to use PETs in recommender systems is computational cost. In our experiments, using Federated Learning caused a 1.3× training time increase over the baseline centralized model. This cost arises from performing many local training iterations and securely aggregating model updates instead of simply centralized optimization. When Differential Privacy was added on top of Federated Learning, the computational overhead was infinitesimally small, merely 5% over and above the cost, as adding noise is computationally lightweight compared to training deep learning models. When Homomorphic Encryption was used with Federated Learning, the computational overhead became significantly more evident. Training time doubled, resulting in 2.1× the runtime compared to the baseline.

This follows the shortcomings of Homomorphic Encryption, whereby huge resources must be allocated to conduct operations over encrypted values. A comparison of computation overhead across different methods is provided. The discussion points out that though Federated Learning and Differential Privacy bring in computationally manageable overheads suitable for most commercial deployments, Homomorphic Encryption integration requires serious consideration, especially in resource-limited settings.

6. Conclusion

PET adoption in recommender systems is a major step towards achieving the best of both worlds, P-RECS. The experimental study conducted on the MovieLens 1M dataset substantiates the claims of the PETs like federated learning, differential privacy, and homomorphic encryption implementation impact on the recommendation system with insignificant performance loss. Namely, the changes in accuracy varied across different privacy-preserving models while remaining below 5%, meaning that users can continue receiving accurate and valuable recommendations for private data. Also, the above-given PETs brought down the measured privacy risk by a percentage of about 80%.

They increased the ability of the system to prevent adversary attacks on data privacy, which is essential in meeting the requirements of contemporary privacy regulations such as GDPR and CCPA. Besides the effectiveness of the recommendation quality, we identify that privacy-preserving models can still scale commercially viable in practice, especially when employing Federated Learning and Differential Privacy. The full Homomorphic Encryption computation mode has a big computational overhead. Hence, a fully encrypted scheme may not be suitable for certain applications with stringent time or spatial requirements. Overall, this research indicates that PETs are not only conceptual artefacts but technologies that can enhance the privacy protection offered by recommender systems while making efficiency gains.

6.1 Future Work

However, as highlighted in this study, there are several directions the research can take for the integration of PETs into recommender systems. One is the improvement of the overhead of Homomorphic Encryption schemes so that fully encrypted recommendation computations become efficient even on small devices such as smartphones and IoT systems. Further to this, future research directions include incorporating lightweight privacy-preserving approaches, for instance, integrating other forms of FL like Differential Privacy and Secure Aggregation or even higher efficiency than those used above. , as well as enable work on extending the validity of designed approaches to the given problem beyond the NLTK to other actual and frequently explored domains like social media activity, purchase histories, and multimedia consumption.

Reference

[1] Ghosh, A., Roughgarden, T., & Sundararajan, M. (2009, May). Universally utility-maximizing privacy mechanisms. In Proceedings of the forty-first annual ACM symposium on Theory of computing (pp. 351-360).

[2] Himeur, Y., Sohail, S. S., Bensaali, F., Amira, A., & Alazab, M. (2022). Latest trends of security and privacy in recommender systems: a comprehensive review and future perspectives. Computers & Security, 118, 102746.

[3] Shi, E., Chan, H. T. H., Rieffel, E., Chow, R., & Song, D. (2011). Privacy-preserving aggregation of time-series data. In Annual Network & Distributed System Security Symposium (NDSS). Internet Society.

[4] McMahan, B., Moore, E., Ramage, D., Hampson, S., & y Arcas, B. A. (2017, April). Communication-efficient learning of deep networks from decentralized data. In Artificial intelligence and statistics (pp. 1273-1282). PMLR.

[5] Dwork, C. (2008, April). Differential privacy: A survey of results. In International conference on theory and applications of models of computation (pp. 1-19). Berlin, Heidelberg: Springer Berlin Heidelberg.

[6] Aono, Y., Hayashi, T., Wang, L., & Moriai, S. (2017). Privacy-preserving deep learning via additively homomorphic encryption. IEEE transactions on information forensics and security, 13(5), 1333-1345.

[7] Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., ... & Seth, K. (2017, October). Practical secure aggregation for privacy-preserving machine learning. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (pp. 1175-1191).

[8] What are privacy-enhancing technologies (PETs)? decentriq, online. https://www.decentriq.com/article/what-are-privacy-enhancing-technologies

[9] Shokri, R., & Shmatikov, V. (2015, October). Privacy-preserving deep learning. In Proceedings of the 22nd ACM SIGSAC conference on computer and communications security (pp. 1310-1321).

[10] Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., & Zhang, L. (2016, October). Deep learning with differential privacy. In Proceedings of the 2016 ACM SIGSAC conference on computer and communications security (pp. 308-318).

[11] Duchi, J. C., Jordan, M. I., & Wainwright, M. J. (2014). Privacy-aware learning. Journal of the ACM (JACM), 61(6), 1-57.

[12] Shokri, R., Stronati, M., Song, C., & Shmatikov, V. (2017, May). Membership inference attacks against machine learning models. In 2017 IEEE Symposium on Security and Privacy (SP) (pp. 3-18). IEEE.

[13] Papernot, N., Abadi, M., Erlingsson, U., Goodfellow, I., & Talwar, K. (2016). Semi-supervised knowledge transfer for deep learning from private training data. arXiv preprint arXiv:1610.05755.

[14] Dwork, C., & Roth, A. (2014). The algorithmic foundations of differential privacy. Foundations and Trends® in Theoretical Computer Science, 9(3–4), 211-407.

[15] Fan, L., & Xiong, L. (2013). An adaptive approach to real-time aggregate monitoring with differential privacy. IEEE Transactions on knowledge and data engineering, 26(9), 2094-2106.

[16] Lye, G. X., Cheng, W. K., Tan, T. B., Hung, C. W., & Chen, Y. L. (2020). Creating personalized recommendations in a smart community by performing user trajectory analysis through social Internet of Things deployment. Sensors, 20(7), 2098.

[17] Vaidya, J., & Clifton, C. (2003, August). Privacy-preserving k-means clustering over vertically partitioned data. In Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining (pp. 206-215).

[18] Yao, A. C. (1982, November). Protocols for secure computations. In 23rd annual symposium on foundations of computer science (sfcs 1982) (pp. 160-164). IEEE.

[19] Fredrikson, M., Jha, S., & Ristenpart, T. (2015, October). Model inversion attacks that exploit confidence information and basic countermeasures in Proceedings of the 22nd ACM SIGSAC conference on computer and communications security (pp. 1322-1333).

[20] Nissim, K., Raskhodnikova, S., & Smith, A. (2007, June). Smooth sensitivity and sampling in private data analysis. In Proceedings of the thirty-ninth annual ACM symposium on Theory of computing (pp. 75-84).