IEEE Open Journal of the Computer Society

Received 14 February 2024; revised 29 April 2024; accepted 29 April 2024. Date of publication 2 May 2024; date of current version 15 May 2024. The review of this article was arranged by Associate Editor Jingcai Guo.

Digital Object Identifier 10.1109/OJCS.2024.3396344

A Survey on the use of Federated Learning in Privacy-Preserving Recommender Systems

CHRISTOS CHRONIS [1], IRAKLIS VARLAMIS [1] (Member, IEEE), YASSINE HIMEUR [2] (Senior Member, IEEE), AYA N. SAYED [3], TAMIM M. AL-HASAN [2], ARMSTRONG NHLABATSI [3], FAYCAL BENSAALI [2] (Member, IEEE), AND GEORGE DIMITRAKOPOULOS [1] (Member, IEEE)

[1]Department of Informatics and Telematics, Harokopio University of Athens, 176 76 Kallithea, Greece
[2]Department of Electrical Engineering, College of Engineering, Qatar University, Doha 2713, Qatar
[3]KINDI Computing Research Centre, College of Engineering, Qatar University, Doha 2713, Qatar

CORRESPONDING AUTHOR: CHRISTOS CHRONIS (e-mail: chronis@hua.gr).

This work was supported by the National Priorities Research Programme (NPRP) under Grant NPRP14S-0401-210122, from the Qatar National Research Fund (a member of The Qatar Foundation).

ABSTRACT In the age of information overload, recommender systems have emerged as essential tools, assisting users in decision-making processes by offering personalized suggestions. However, their effectiveness is contingent on the availability of large amounts of user data, raising significant privacy and security concerns. This review article presents an extended analysis of recommender systems, elucidating their importance and the growing apprehensions regarding privacy and data security. Federated Learning (FL), a privacy-preserving machine learning approach, is introduced as a potential solution to these challenges. Consequently, the potential benefits and implications of integrating FL with recommender systems are explored and an overview of FL, its types, and key components, are provided. Further, the privacy-preserving techniques inherent to FL are discussed, demonstrating how they contribute to secure recommender systems. By illustrating case studies and significant research contributions, the article showcases the practical feasibility and benefits of combining FL with recommender systems. Despite the promising benefits, challenges, and limitations exist in the practical deployment of FL in recommender systems. This review outlines these hurdles, bringing to light the security considerations crucial in this context and offering a balanced perspective. In conclusion, the article signifies the potential of FL in transforming recommender systems, paving the path for future research directions in this intersection of technologies.

INDEX TERMS Recommender systems, federated learning (FL), privacy-preserving techniques, distributed learning.

I. INTRODUCTION

In the vast realm of the digital world, Recommender Systems (RSs) serve as compasses, guiding users through the overwhelming array of choices to find what truly suits their tastes and preferences. As the architects of personalized user experiences, these intelligent systems have become the backbone of many thriving online platforms, from e-commerce giants like Amazon to streaming powerhouses such as Netflix [1]. RSs utilize sophisticated algorithms to analyze a user's behavior, preferences, and interactions. They deftly sift through massive datasets to spot patterns and connections, enabling them to predict and suggest items a user would likely be interested in. Whether it's a book, movie, song, or product, RSs bring users and their desired items closer together in a seamless combination of satisfaction and discovery [2]. The importance of RSs in the digital landscape cannot be overstated. They enable businesses to provide personalized experiences at scale, boosting user engagement, enhancing customer satisfaction, and driving business growth. They turn the seemingly impossible task of understanding each individual user's preferences into an achievable goal, creating a win-win situation for both businesses and users [3]. Beyond personalization, RSs also play a crucial role in information discovery. They broaden our horizons by suggesting items we may never have found on

our own, thereby enriching our digital experiences. From suggesting a new genre of music to introducing us to an emerging author, RSs empower us to explore and discover in exciting and unexpected ways.

As RSs become increasingly integral to our online experiences, concerns surrounding privacy and security have escalated. These systems handle an enormous amount of user data, including preferences, browsing history, and personal details, making them prime targets for data breaches [4]. The unauthorized exposure of this sensitive information could lead to severe privacy violations, causing substantial harm to individuals. Additionally, sophisticated RSs often employ complex machine learning algorithms, which further obfuscate data processing, leading to a lack of transparency and potential misuse. Critics also express concerns about manipulating recommendations, when certain items are prioritized over others due to hidden algorithms or paid promotions, thus leading to biased results [5]. The increasing use of these systems in sensitive areas such as healthcare elevates the potential risks associated with privacy breaches and biased recommendations. It is therefore imperative for companies to implement robust security measures, improve transparency, and establish stringent data ethics guidelines to ensure the trust and safety of users while interacting with RSs.

Federated Learning (FL) emerges as a game-changing privacy-preserving approach that addresses prevailing data security and privacy concerns in machine learning applications. Unlike traditional centralized models that transfer and process raw data centrally, FL allows models to be trained locally on the user's device. The system is trained in a decentralized manner on the data sources, and only model updates are sent to a central server for aggregation, drastically reducing the need for data transmission. Consequently, the risk of data leakage and breaches are minimized. This paradigm not only preserves data privacy by keeping sensitive information localized but also reduces reliance on data centralization, mitigating the single point of failure risk. FL can also leverage real-time, personalized user data to improve the quality and relevance of model predictions, enhancing user experiences while preserving privacy. Moreover, reducing data transmission can alleviate network load and save bandwidth. Thus, FL offers a promising solution that covers the needs of advanced machine learning applications and satisfies data privacy requirements.

The advent of FL provides a promising solution for advancing RSs while preserving user privacy. RSs, pivotal in tailoring user-specific content, depend on large-scale user data for their operations, raising serious privacy and security concerns. The distributed machine learning approach introduced by FL ensures that data never leaves the user's device, mitigating privacy risks. The process of training local models on each device and exchanging model updates only has two-fold benefits: it alleviates the need to aggregate and centrally process a vast amount of user data, thus reducing data storage and transmission costs, and it maintains data privacy by default as sensitive user data never leaves the local device. In an era where data privacy is a growing concern, incorporating FL in RSs offers an effective strategy to harness the power of data while respecting user privacy. By advancing RSs with FL, we can strike a balance between personalization and privacy, which is crucial for gaining user trust and enhancing the overall user experience.

A. RESEARCH QUESTIONS

This work aims to delve into the synergies between FL and RSs, envisioning a future where personalized recommendations are not only accurate but also protective of user privacy. We believe that the integration of FL into RSs has the potential to redefine the landscape of user-centric experiences. For this purpose, we perform an overview of existing solutions in RSs and Federated Machine Learning and describe their main components and internal mechanisms. We explain how FL and RS can be successfully combined and highlight the potential benefits of combining FL with RSs. Finally, we examine the security concerns that may still arise, and discuss the various privacy-preserving techniques that can be applied in the FL example to further enhance data privacy. The main Research Questions that arise for this study are:

- *RQ1:* What are the main approaches that allow the use of FL in the recommendation task?
- *RQ2:* Is FL capable of addressing the security and privacy concerns of RSs?
- *RQ3:* What are the open challenges for using FL in modern RSs?

Consequently, we survey the field to elucidate how this fusion can balance between enhancing user satisfaction through tailored recommendations and safeguarding sensitive user data at its core. Ultimately, we chart a roadmap toward the successful combination of recommendation accuracy and user privacy that will foster trust, empowerment, and an unparalleled digital experience.

B. RELATED SURVEYS

A search in the related literature for survey works on federated recommender systems revealed the work of Harasic et al. [6], which provides the main privacy-preserving techniques of FRSs, namely encryption, perturbation, and masking. The work aims to cover all current and future challenges in FRSs and provides only a few details on the privacy aspect. The latest survey of Sun et al. [7] also surveys the various FedRS communication architectures, without giving much emphasis on the basics of Federated Learning, lists the main techniques for privacy protection, including homomorphic encryption, differential privacy, etc., and the main attack types. The article also examines the communication cost aspect and provides a list of generic datasets widely used for training and evaluating recommender systems. The earlier survey of Yang et al. [8] provides a more detailed explanation and a formal definition of Federated Learning in Recommender Systems and lists the various architectural alternatives. The work also provides future challenges and research directions but does not list any applications or task-specific datasets. Finally, the

TABLE 1. A Comparison of the Most Recent Survey Works in the Field and Their Features

Work	[6]	[7]	[8]	[9]	Proposed work
Need establishment					✓
Paper selection methodology					✓
Data analysis					✓
FedRS basics	✓	✓	✓		✓
FL Architectures	✓	✓	✓		✓
Privacy techniques	✓	✓	✓	✓	✓
Applications	✓	✓		✓	✓
Datasets	✓	✓		✓	
Evaluation metrics					✓
Challenges	✓	✓	✓	✓	✓
Future Directions	✓	✓			✓

work of Asad et al. [9] provides a more broad overview, without providing many details on the FedRS mechanisms and the available architecture alternatives. They emphasize privacy-preserving techniques, provide a list of applications and general-purpose datasets, and discuss the main challenges which are also considered future research directions.

In the current work, we attempt to combine the merits of previous surveys based on a methodological selection of the most relevant research works in the field, and on a set of research questions that had to be answered. This methodology was missing from previous surveys. In addition, as shown in Table 1 the current work performs an analysis of the works found in the literature, before proceeding with the presentation of architectures, applications, or challenges. We purposely avoid repeating information about the available datasets, since the current datasets are generic and not specially designed for Federated Learning and privacy, but we provide a discussion on the metrics that can be used for evaluating the performance of the proposed solutions.

C. CONTRIBUTIONS

Compared to the previous works in the field, this article:

- performs an extended overview of FRSs that focus on the privacy aspect,
- jointly examines FL and RSs and provides a reference to real-world implementations or applications,
- discusses the main challenges and limitations of existing approaches and identifies the areas of future research.

Our survey follows a conceptual approach that begins with analyzing the various techniques employed by recommender systems and then explains their main privacy challenge, which stems from the centralized manner of processing user preferences. Consequently, it introduces the Federated Learning approach as a solution to the security and privacy challenges of RSs and illustrates the various architectural alternatives. The approach explains the various privacy-preserving techniques used in FedRS and presents the benefits of their use in different application domains. However, it also sheds light on the security risks that may still lurk for FedRS and discusses methods that can be used to avoid them, as part of future research. The conceptual path of our survey is presented in Fig. 1.

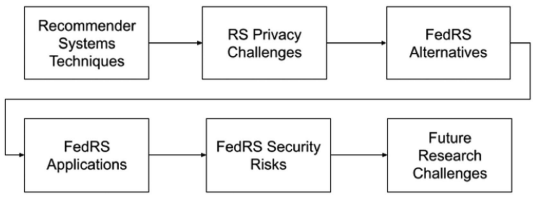

FIGURE 1. The conceptual path followed in the survey.

The key contributions of the current survey are:

1) *Introduction of Federated Learning (FL) as a Privacy-Preserving Solution*, acknowledging the potential of FL to maintain user privacy while providing effective personalized recommendations.
2) *Exploration of Benefits and Implications of FL in Recommender Systems* and of the consequences of integrating FL with recommender systems, supported by examples and application areas from the literature. This provides a tangible understanding of the potential benefits of real-world applications.
3) *In-Depth Overview of Federated Learning*, covering its types, key components, and privacy-preserving techniques. This section equips readers with a solid understanding of FL and its applicability to enhance the security of recommender systems.
4) Identification of Challenges and Limitations with a Balanced Perspective that brings security considerations to light and contributes to a more detailed understanding of the practical implications and potential hurdles in the integration of FL with RS.

Section II that follows, provides a detailed analysis of the methodology we employed to collect pertinent literature, and Section III provides a high-level analysis of the search results. The following sections perform a deeper analysis of the current research in the field. Section IV presents various types of RS and in Section V, discusses the challenges encountered by a typical RS, especially about security issues. Section VI introduces FL and describes the standard architectures of an FL system. Section VII focuses on the combination of FL and RS, exploring their collaborative potential and associated security risks. Section VIII explores how a FedRS can be utilized across different domains. The emphasis of Section IX is on the limitations and challenges of a FedRS, with a special focus on security aspects, whereas Section X aims to address the research questions raised earlier in the paper. Finally Section XI summarizes our findings and proposes directions for future research.

II. METHODOLOGY

This methodology section, aligned with PRISMA 2020 guidelines, outlines the systematic approach adopted for reviewing

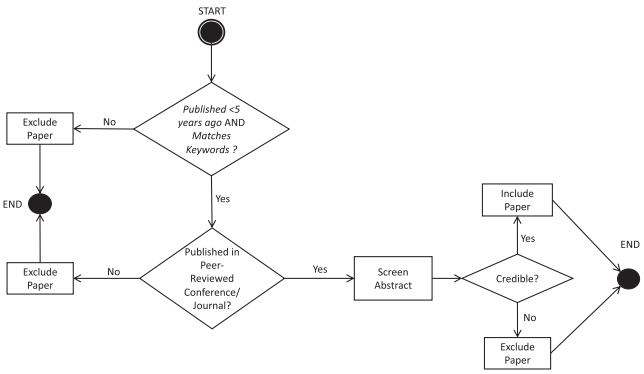

FIGURE 2. Flowchart illustrating how papers were systematically selected for the survey.

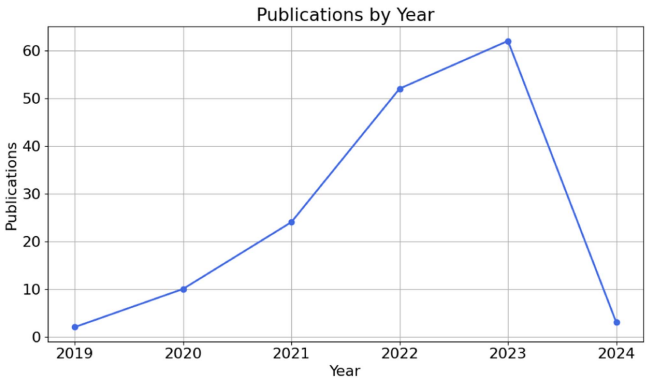

FIGURE 3. This diagram illustrates the documents published from 2019 until 2024.

literature on Federated Recommender Systems (FedRSs), with a particular focus on their security and privacy aspects.

- *Search Strategy:* The literature search was conducted across academic databases, and more specifically on Google Scholar, Scopus, and IEEE Xplore. Scopus has been used first to get an idea of the volume of publications per year and the domains they come from. The search parameter employed in Scopus was *"TITLE-ABS-KEY ("federated learning" AND "recommender systems" AND (privacy OR security)) AND PUBYEAR > 2018"*. This strategy was tailored to capture publications that primarily focus on *"Federated Learning"* and *"Recommender Systems"*, examining their *"security"* or *"privacy"* aspects. This approach is intended to be both easily duplicated and thorough and to provide an overview of the publication landscape. Its results are further discussed in Section III.
- *Search Criteria:* Consequently, the search was extended to Google Scholar and IEEE Xplore, and in addition to the aforementioned keywords, more composite phrases like "federated learning in recommender systems", "privacy-preserving techniques in Federated Recommender Systems", and "security challenges in Federated Recommender Systems" were used. More similar queries and manual validation of the results have been performed to acquire more related papers, including pre-prints. By screening the abstracts, assessing their credibility, and prioritizing the peer-reviewed sources we finally selected articles that distinguish the different FL types, their components, and the privacy-preserving techniques that apply in FL. Employing a snowballing technique, references from selected articles are explored, while iterative refinement helped us adjust our search strategies. Fig. 2 illustrates the process we followed in selecting papers for inclusion in the survey. The results of our bibliography research are presented in the sections that follow.
- *Inclusion and Exclusion Criteria:* The search was restricted to English-language articles. We limited the search to works that have been published in the last five years and further filtered the papers using some more criteria. The inclusion criteria were: Peer-reviewed articles focus on FedRSs, their privacy-preserving techniques, and various aspects such as training speed, parallelization, synchronization, and user control over data sharing. Exclusion criteria included non-peer-reviewed articles, and publications not specific to FedRSs or federated learning.
- *Quality Assessment:* The quality of the selected papers was evaluated based on their relevance, methodological rigor, the impact factor of the journals, and citation counts, ensuring the inclusion of credible and significant studies.
- *Data Extraction and Synthesis:* Data extraction focused on identifying categories of FedRSs, privacy-preserving techniques, and various aspects of FedRSs, including those that do not directly address privacy concerns. We specifically considered survey works illustrating different categories of FedRSs, studies discussing aspects like training speed [10], parallelization [11], [12], user control [13], and works emphasizing privacy through secure computation and communication protocols [14], [15], [16], [17], [18].
- *Results of the Literature Search:* The findings are organized in alignment with the document's structure, providing a detailed analysis of FedRSs, federated learning, and their intersection with security and privacy considerations. This includes examining the existing literature that highlights the importance and various dimensions of FedRSs.

III. DATA ANALYSIS

In Fig. 3 we can see the trend in the number of publications clearly shows a growing interest in the subject. Starting with just 2 papers in 2019, there's been a steady increase each year. By 2020, there were 10 papers, and this number jumped to 24 in 2021, and even more to 52 by 2022. The highest spike was in 2023 with 62 papers. Although 2024 shows only 3 papers so far, this is probably because the year is still ongoing. This pattern points out how more and more researchers are

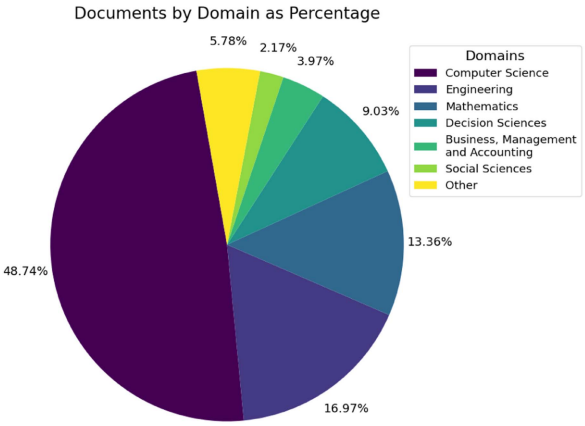

FIGURE 4. This diagram illustrates the documents published by domain.

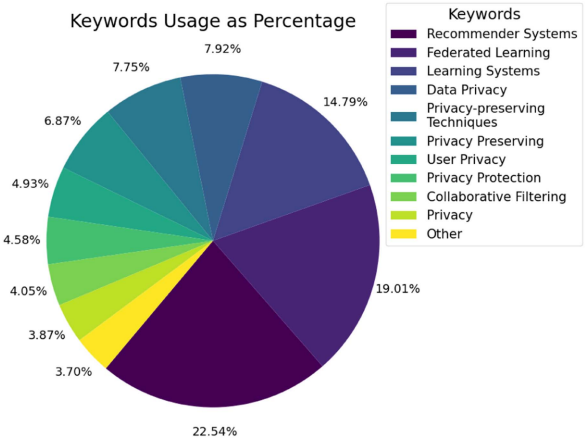

FIGURE 5. This diagram illustrates the top 10 most common keywords between the publications.

getting interested in this area, showing its rising importance in various fields.

The analysis of data extracted from Scopus as presented in Fig. 4 reveals a significant trend in the distribution of research across various domains. The majority of the studies, amounting to 182 out of 153 total studies, are centered in the fields of Computer Science and Engineering. This represents approximately 66% of the total research output. However, it is noteworthy that the subject also gains interest from a range of other fields. For instance, there are notable contributions from Mathematics (13.36%), Decision Sciences (9.03%), and Business Management and Accounting (3.97%). Additionally, the presence of research from domains such as Social Sciences, Medicine, Materials Science, and Energy, although in smaller numbers, underscores the subject's cross-domain attraction and its significance in various fields. Such a varied academic landscape indicates that while the majority of research is rooted in Computer Science and Engineering, there is a growing recognition of the subject's applicability and significance in other disciplines.

In Fig. 5, the top ten keywords among the publications are presented, highlighting the focal themes and areas of

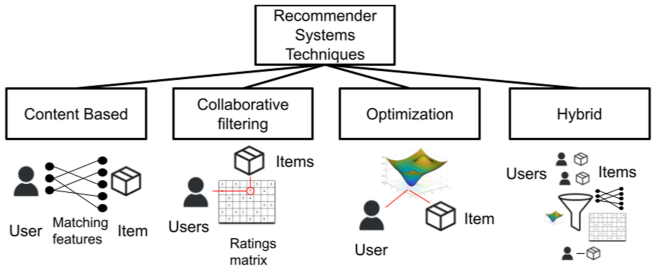

FIGURE 6. The different techniques used in RSs.

interest in the research. This analysis filters out keywords that appeared more than 20 times across publications. Notably, 'Recommender Systems' leads the list with a frequency of 22.54%, underscoring its prominence in the current research landscape. It is closely followed by 'Federated Learning', which holds a significant frequency of 19.01%. Keywords occurring less frequently have been consolidated under the 'Other' category. This category, with a collective frequency of 3.7%, encompasses a diverse array of topics directly related to the field (e.g. Matrix Factorization, Differential Privacy, Learning Algorithms, Data Models, etc.). Such categorization aids in emphasizing the most influential themes in current research while acknowledging a broader spectrum of relevant but less dominant topics.

IV. RECOMENDER SYSTEMS: AN ANALYSIS

Recommender Systems assume several users who access a large pool of items and try to limit for each user the list of items of potential interest, rank them, and propose the highly ranked to the user accordingly. The vast number of RS approaches can be broadly grouped into three main types: content-based, collaborative filtering and optimization-based [19]. Fig. 6 categorizes these approaches and illustrates their distinct methodologies.

Content-based RSs [20] suggest items to users based on the characteristics and attributes of the items and the user's previous preferences. They rely on the content or features of items, such as their text descriptions, keywords, or metadata, and recommend items similar in content to those the user has shown interest in. For example, if a user has previously liked action movies, a content-based RS might recommend other action movies with similar themes, actors, or descriptions.

Collaborative filtering (CF) RSs [21] leverage the behavior and preferences of a group of users to identify patterns and similarities in user behavior, such as item ratings, purchases, or clicks, and consequently recommend items to a user based on the preferences of users with similar tastes. The two main CF approaches, namely user-based and item-based, suggest items preferred by users with similar tastes and items similar to those preferred by the user, respectively.

Optimization-based RSs [19] use mathematical and computational techniques to find the most suitable items to recommend to users. They explicitly model and solve optimization problems, such as maximazing user satisfaction or

engagement, while considering constraints or objectives, such as diversity, novelty, or fairness in recommendations. They provide a powerful framework for fine-tuning recommendation algorithms, allowing incorporating complex business rules and preferences. These approaches have been beneficial in large-scale, real-world RSs where personalized and context-aware recommendations are essential for enhancing user experiences and achieving business goals.

Finally, hybrid RSs combine multiple recommendation techniques to provide more accurate and diverse recommendations. By blending different methods, hybrid RSs aim to offer a more personalized and effective recommendation experience. They either act as meta-systems and combine recommendations from different models, or incorporate user feedback to re-rank or filter recommendations, depending on the specific application and goals.

Traditional RSs excel at analyzing user behavior, preferences, and interactions and thus enhance user satisfaction and engagement. They allow the introduction of new and unexpected items, encouraging exploration and discovery, while saving precious time from searching or browsing huge item catalogs. Their ability to handle millions of users and hundreds of thousands of items makes them suitable for a wide range of applications, from e-commerce to content streaming services.

Despite their advantages, RSs have several limitations [22] that are considered by researchers and solution implementers. For example, although they perform well for known users and items, they struggle to provide accurate recommendations for new users or items with little to no historical data. This problem, known as *cold start* can result in poor user experiences. On the other side, although they scale well on large datasets, they still rely on user-item interaction data to create user profiles. *Data sparsity* is evident in applications where users only rate a few from the thousands of available items, and this can lead to inaccurate recommendations. The *lack of transparency* and *inability to adapt to preference changes* are two additional problems, that can affect trust and user satisfaction, whereas the *popularity bias* may also lead to a limited set of items that keep being recommended and other niche items being overlooked.

In addition to the aforementioned issues, several security and *privacy concerns* also apply to modern RSs that collect and store user behavior data for recommendation purposes. If this data is not adequately protected, it may be vulnerable to breaches and misuse. The impact of RSs on item popularity makes them the target of data poisoning attacks, from malicious users or entities who manipulate the system's recommendations by injecting fake or biased data into the user-item interaction history. Data encryption and the use of blockchain have been recommended as a solution to maintain the integrity of RSs [5]. Even when the user data are private, there is still the risk of inferring implicit user attributes (e.g., their preferences) by analyzing the recommendations provided by the system, potentially compromising user privacy. Model-based attacks performed by adversaries may result in the manipulation of the underlying recommendation algorithms to promote specific items or target specific users.

To address these limitations and security concerns, hybrid approaches that combine deep learning-based methods and encryption, have been developed to provide more accurate and privacy-preserving recommendations while mitigating some of the drawbacks associated with traditional systems. The main techniques studied extensively in the literature focus on privacy-preserving collaborative filtering and comprise cryptographic techniques (e.g., homomorphic encryption) and randomization techniques (e.g., Grabled Circuits) [23]. Fully or partially homomorphic encryption of Matrix Factorization and Ridge Regression techniques have been proposed to learn item profiles without exposing user preferences. More details on the aforementioned techniques are provided in the section that follows.

V. SECURITY CONSIDERATIONS AND PRIVACY-PRESERVING RECOMMENDER SYSTEMS

In the age of data-driven decision-making and personalized user experiences, RSs have emerged as a cornerstone of modern digital platforms. These systems, often fueled by advanced machine learning algorithms, play a pivotal role in guiding users to discover content, products, and services that align with their preferences and needs [24]. However, while their contributions to enhancing user engagement and satisfaction are undeniable, they also raise significant concerns regarding user privacy and system security [25].

The advent of RSs has transformed how we interact with digital environments. Yet, as these systems gather and process vast amounts of user data to deliver personalized recommendations, they become enticing targets for malicious actors seeking to exploit vulnerabilities and compromise the confidentiality, integrity, and user information availability [25], [26]. Therefore, addressing security considerations in privacy-preserving RSs has become imperative for developers, data scientists, and organizations alike [25].

The need for Privacy-Preserving Personalized RSs was clearly stated in a comprehensive literature survey that highlighted the main privacy challenges of existing RSs and the works that focus on privacy protection in related tasks [23]. This section delves into the multifaceted realm of security considerations within the context of privacy-preserving RSs. It explores the complex interplay between pursuing personalized user experiences and safeguarding user privacy and system integrity. It discusses the potential risks and consequences of security and privacy challenges and evaluates the vulnerabilities and threats of RSs.

A. SECURITY CHALLENGES AND SOLUTIONS IN RECOMMENDER SYSTEMS

Traditional RSs are susceptible to various security challenges, including data breaches and privacy violations [26]. The distributed nature of RSs, involving multiple parties and data sources, increases the risk of privacy violations. The potential risks and consequences of security challenges in RSs are

significant. Data breaches and privacy violations can lead to the loss of user trust and damage to the reputation of the platform or organization [26]. When users perceive that their privacy has been breached, they may become reluctant to share personal information or engage with the RS, resulting in increased data sparsity and loss of user engagement and satisfaction. Moreover, privacy violations can have legal and regulatory implications, leading to financial penalties and legal consequences for the platform or organization [27].

RSs are vulnerable to various security vulnerabilities and threats. One common vulnerability is manipulating user data or recommendations by malicious actors [28]. Shilling attacks, for example, involve the insertion of fake or biased ratings to manipulate the recommendation algorithm and promote specific items or agendas [28]. These attacks can compromise the fairness and integrity of the RS, leading to inaccurate and biased recommendations [29]. Algorithms for detecting and preventing attacks have been proposed, such as shilling attack detection algorithms, to identify and mitigate the impact of malicious ratings on the recommendation process [29]. Similarly, secure computation mechanisms, such as secure inner product computation, have been utilized to support user queries without compromising privacy [30].

Another vulnerability is the breach of privacy and anonymity, which can occur when information is transmitted or personal identification is compromised [31]. This breach of privacy can result in the unauthorized disclosure of sensitive user information and the inference of additional personal details [31]. One approach to privacy-preserving RSs is the use of collaborative filtering for implicit feedback datasets. Collaborative filtering is a technique that identifies preference-based user groups and makes recommendations based on the memberships of users in these groups. Consequently, the collaborative filtering approach allows for personalized recommendations without directly exposing individual user preferences.

Privacy-preserving recommendation algorithms that build on distributed matrix factorization have been proposed in the literature [32] in order to train local models, which are then exchanged in the form of latent factors with nearby users. Encryption and access control have also been proposed as privacy-preserving mechanisms to protect user data and ensure confidentiality. Advanced encryption techniques, such as Attribute-Based Encryption (ABE), can be employed to ensure end-to-end privacy in cloud storage systems [26]. Fully homomorphic encryption has been proposed in [33] to obfuscate the stated user preferences without harming the accuracy of the Matrix-Factorization-based RS. Similarly, homomorphic encryption has been employed to perform computations on encrypted data without compromising the RS performance [25].

VI. FEDERATED LEARNING: AN OVERVIEW

FL is a technique where multiple parties collectively train a machine learning model without the need to share raw data either among themselves or with a central server. This approach combines techniques from various domains, including distributed systems, machine learning, and security [34]. The primary advantage of FL lies in its ability to preserve the privacy of each participant's individual data.

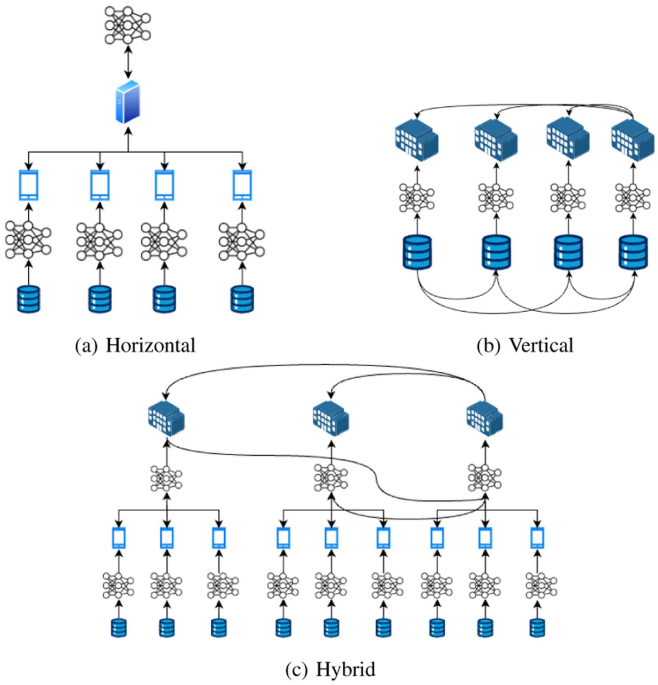

FIGURE 7. The distinct types of FL systems, differentiated by data distribution and aggregation techniques.

Based on the similarities or diversities observed in the exposed data of each participant of parties, every participant generates a local model, which can either differ or exhibit substantial similarities with models from other parties. Eventually, a new global model is produced by aggregating all the individual models using various techniques such as Federated Averaging (FedAvg) [35] or FedFast [10]. The performance of the new global model is expected to surpass that of the individual parties models, as evaluated by a chosen metric such as accuracy on test data. It is essential for all participants to employ a common architecture to facilitate this process.

A. FL TYPES

The FL systems can be categorized into three types (horizontal, vertical, and hybrid FL) based on the data distribution and how they perform their aggregation actions. Fig. 7 provides a visual representation of these different FL types.

The **horizontal FL** approach is commonly adopted when various parties aim to improve a shared task (e.g., classification). This method involves training individual local models separately, which are then amalgamated into a global model using techniques like FedAvg [35]. Frequently applied in cross-device settings, this approach necessitates that all models possess the same structure and operate within the same feature space.

The **vertical FL** approach is a method where various parties have different feature spaces, yet there's a minor overlap among them. This approach is typically seen among different companies that collect user data (features or entities) for diverse purposes, where some features might be common. In this method, the most prevalent techniques include entity alignment techniques between the entity segments that come from different parties, as referenced in [36]. The overlapping sets of features are then encrypted and transmitted to contribute to the training of the global model.

Finally, depending on the application and available data, **hybrid methods** can also be employed. A notable example of this can be seen in cancer diagnosis using data shared among different hospitals. A potential technique that could be utilized is Transfer Learning, as outlined in source [37], executed securely as proposed by [38].

B. COMPONENTS

Every FL system comprises three fundamental components: a group of parties, a manager, and a framework that handles communication and computations to generate the new aggregated model.

The parties in an FL system serve as the data owners and the ultimate consumers of the newly produced model. They often have limited computational resources and may also face additional limitations, such as power constraints or communication issues, especially in cross-device settings, which involve parties using devices of different computational capacities. The parties can also include organizations (e.g., groups of hospitals in an FL predictor that employs patient data) that contribute the data of their users and are referred to as cross-silo participants [39]. Two important considerations in the design of an FL system are scalability and stability, with notable differences between cross-silo and cross-device settings. In a cross-silo setting, there is generally greater stability, making it easier to handle more demanding computational and communication tasks [36]. On the other hand, in a cross-device setting, stability is often compromised due to factors such as mobile devices with poor connections [40] or limited computational resources. However, scalability is more feasible in this setting by utilizing a larger number of devices. Additionally, due to the limited availability of parties, it is common practice to involve only a fraction of the available devices in each computational round [35], [41]. Furthermore, a common challenge affecting both cross-silo and cross-device settings is data distribution among the parties. The data distribution is often non-IID (identically and independently distributed) [39], which has been extensively investigated in various studies [42], [43], [44], [45].

The manager can vary based on the parties' setting. In the cross-device setting, a manager is a powerful central server capable of handling the computational needs of the model aggregation. If the server is not stable and reliable, it can degrade the performance of the entire FL system and lead to bad models. In recent years, the usage of blockchain [46] has been introduced to increase the system's reliability. In contrast, in the cross-silo setting, the manager can be anyone from the parties, and in most cases is the node that has the most available resources in each round. When the participant nodes are capable enough to perform computationally heavy tasks, the vertical FL [34] approach is preferred. Although the resulting setting is fully decentralized, with parties communicating with each other and exchanging models, its design is very challenging due to the significant communication overhead it creates [42].

A crucial part of FL Systems (FLSs) is the underlying mechanisms responsible for the **communication** and the **computation/aggregation**. The popular aggregation framework of FedAvg [35], involves the server that disseminates the updated global model to all parties, who adjust the model with their respective local data. These updates are returned to the server, which averages the input to yield a new global model. The process repeats until a pre-established number of iterations is reached, with the server's global model as the final output. An alternative framework is SimFL [16], a decentralized framework in which each participating party first updates the gradients based on their local data, and then sends the updated model only to a single party. That node uses its local data and received gradients to refine its own model, which is then disseminated to other parties, and so on. The model update process is designed to be fair, allowing every party an equal opportunity to update the model, with the final model determined after a fixed number of iterations.

C. PRIVACY-PRESERVING TECHNIQUES IN FEDERATED LEARNING

FL is known to be a safe method when we talk about privacy. This is mainly because local data is not shown to other groups, and the whole process is built on swapping models between them [47], [48], [49], [50]. However, several attacks can happen on machine learning models, including model inversion attacks [47] and membership inference attacks.

Many of these attacks try to find out raw data from the model that has been exchanged. To fight these attacks, researchers have come up with privacy tools like differential privacy [51] and k-anonymity. These privacy tools can be broadly grouped into two groups: cryptographic methods and differential methods.

Cryptographic methods, such as homomorphic encryption [52], [53] and Secure Multi-party Computation (SMC) [54], are very popular [55]. They work by having each person encrypt (turn into a secret code) and decrypt (turn back from the secret code) their messages during the exchange. This method provides a good level of protection in an FLS. However, it doesn't promise perfect safety. For example, in the SMC method, we can't guarantee the final model's safety, and it's still open to inference attacks. In addition, these methods can make the FLS work harder and this extra effort might be a lot, depending on the used cryptographic method.

Differential privacy methods [51], [56] work by trying to limit the effect of a single record on the decisions of a

TABLE 2. Comparison of Federated Recommendation System Algorithms

Category	Communication & Structure	Features & Learning Approach	Drawbacks	Citations
Client-Server	Centralized server, Continuous parameter aggregation	Central server facilitates aggregation, Varies across all learning approaches	Server failure, Privacy concerns	[66], [67]
Peer-to-Peer	Decentralized, Node-to-node, Cross-device/silo	Robustness, Reduced privacy risks, Supports complex models like DL, Meta-learning	Increased computational costs	[68], [69], [70]
Cross-device	Flexible; Client-Server to Peer-to-Peer, Multiple devices with limited resources	Adaptable to resource constraints, Primarily DL and Meta-learning	Varied model complexity due to resource limits	[71], [72]
Cross-silo	Usually Peer-to-Peer; Fewer nodes, high resources	Enhanced data privacy, Advanced models for diverse data	Fewer nodes may limit model diversity	[69], [70]
Matrix Factorization	Suitable for both models, Adapted as needed	User-item interaction, Implicit preference capture	Needs adaptation for varying node capabilities	[73], [67], [74], [71], [72], [75]
Deep Learning	Requires robust model, often Peer-to-Peer, High demand	Complex correlations, Network parameter sharing, Privacy-preserving	High computational demand	[76], [77], [78]
Meta-learning	Adaptable, Peer-to-Peer preferred for privacy, Highly adaptable to limited data environments	Quick generalization, Non-IID data handling, Personalized solutions	High computational demand for training	[79], [80], [81]

TABLE 3. Privacy Levels and Threat Types in FL Techniques

Privacy Level	Threat Type	Description	Citations
Inputs	Data Poisoning Attacks	Parties alter training sample labels to degrade model performance.	[82], [83], [84]
Learning Process	Model Poisoning Attacks	Parties upload specific parameters to decrease global model accuracy.	[85], [86]
	Byzantine Fault	Erratic behavior in parties leading to random updates.	[54], [87]
Learnt Model	Inference Attacks	Sensitive information can be inferred from published models or during the learning process.	[47], [48], [49], [50]

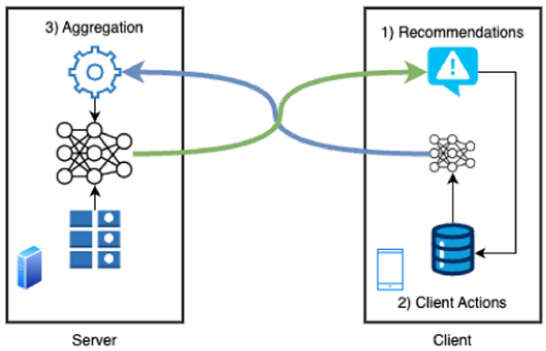

FIGURE 8. The standard training process for FedRS involves a network of devices interfacing with a centralized cloud server.

federated system, so that it becomes difficult to decrypt the record (e.g., the user preference) from the decision (e.g., from the recommendation). Several studies have been proposed in this direction, trying to ensure that the groups don't know whether a record is being used in the learning process [57], [58], [59], [60]. This is usually done by introducing random noise to the data or to the model's parameters [57], [59], [61], thus obfuscating and protecting user records from inference attacks. However, adding noise may have a negative effect on the accuracy of the generated results.

The aforementioned methods can be combined to further improve performance or mixed with other methods, such as the Trusted Execution Environment (TEE), or Intel SGX [62]. The degree of complexity depends on the FLS structure, the desired level of safety, and finally, the effort required for the final system to produce its output [63], [64], [65].

Table 3, provides a summary of the threats that exist in FL techniques, including the distinct types of attacks that may occur at different stages of the learning process.

VII. COMBINING FEDERATED LEARNING WITH RECOMMENDER SYSTEMS

Federated Recommender Systems (FedRS) allow multiple nodes to collaboratively train a recommendation model without exposing or exchanging their actual preferences, simply by exchanging the intermediate parameters of locally trained models. Consequently, the nodes (users) initially train their local models using their actual preferences and then exchange information about the models, aiming to train a final model (global or local versions) that improves the recommendation quality. The performance of the final model is expected to match or exceed that of the models trained locally with the actual user preferences.

A. FEDRS CATEGORIES

To thoroughly understand how the FL approach integrates with RSs, it's essential to first examine the various categorizations of FedRS. Fig. 8 illustrates a typical FedRS pipeline.

Initially, a cloud server generates initial recommendations. Following this, each device independently trains its local learning model and then uploads this model back to the cloud server. In the final stage, the cloud server aggregates these individual models to create an enhanced model and then generates new recommendations. This process is repeated iteratively.

1) BASED ON THE COMMUNICATION MODEL

The exchange of model parameters in FedRS can be performed using two alternative communication architectures: Client-Server and Peer-to-Peer. In the *Client-Server* architecture, parties rely on a trusted central server to aggregate parameters and generate the new global model. This approach is commonly used in Federated Systems in general and FedRS more specifically. The process involves the central server collecting intermediate parameters in each round, performing aggregation, and then transmitting the new global model back to all system parties. This procedure is continuously repeated. The main drawbacks of this architecture comprise the risk of a central server failure impacting the entire system [66], and the potential privacy breaches due to the server's ability to infer client information [67].

The *Peer-to-Peer Architecture* [68], eliminates the need for a centralized server, thus adding to the robustness of the approach. Each node exchanges the parameters of its own local model, only with its neighboring nodes (or some of them), contributing to a locally aggregated model, which is expected to be better than local models and comparable in performance to the global model. This further diminishes the risk of privacy breaches but increases the communication and computational costs for clients.

2) BASED ON THE NODE STRUCTURE

The FedRS systems can be categorized based on the nodes that share their local recommendation models, to Cross-device and Cross-silo (or Cross-platform).

The *Cross-device* approach involves multiple devices that train and exchange local recommendation models. Since some nodes may have limited computational, storage, energy, and communication resources, the local models may vary in complexity or in the frequency of being updated [71], [72].

The *Cross-silo* approach assumes that nodes participate in different platforms (e.g., share preferences in different social networks, or in different shopping sites), usually few in number. The respective federated algorithms aim to improve the overall recommendation performance while respecting privacy and regulatory constraints [69], [70]. Due to limitations in the exchange of actual data, the peer-to-peer communication architecture is usually preferred.

3) BASED ON THE LEARNING APPROACH

Regarding the machine learning approach that is used for generating recommendations, the FedRS approaches are divided into three categories: Matrix factorization, Deep learning, and Meta-learning FedRS.

Matrix factorization-based models are the most prevalent in FedRS [73]. They rely on user-item interaction and rating matrices to capture the implicit user preferences or item properties. User factor vectors are stored locally, and item factor vectors or their gradients are exchanged and aggregated [67], [72], [74], [71], [75].

Deep learning-based models can learn more complex correlations between users and items and have become increasingly popular solutions for RSs. Since their performance is strongly connected to the availability of training data, the privacy concerns that limit data exchange make them suitable for applying testing solutions. Several federated learning approaches have emerged in this field, which attempt to share network parameters among nodes in order to preserve data privacy. Notable studies include the FedNCF framework by Perifanis et al. [76], which involves clients updating and sharing network weights and user/item profiles; FedGNN by Wu et al. [77], using Graph Neural Network models; and a framework by Huang et al. [78] focusing on multi-view recommendation systems. All approaches put emphasis on local processing and collective updating. Despite their excellent performance, these models introduce computational and communication overhead.

Meta-learning is another promising approach for generating recommendations. The ability of Meta-learning models to quickly generalize and learn new tasks [79] makes them appropriate for working in federated setups with limited samples or even using Non-IID (Non-Independent and Identically Distributed) data. In FedRSs, the nodes exchange and aggregate the parameters of their local models in order to learn better and more global models. However, the local models employ locally stored data [80], [81] in their training, and such data are not always IID, which may lead to poor performance. Meta-learning approaches allow the utilization of both the global model and local data to develop personalized solutions.

B. PRIVACY AND SECURITY IN FEDRSS

FL has emerged as a potential solution to address the privacy and security challenges in RSs. By design, the FedRS provide an additional layer of security compared to traditional RSs. This enhanced security is based on the fact that user data are stored in a decentralized manner, and only intermediate parameters (i.e., model gradients) are exchanged with a server node or other nodes. This approach minimizes the risk of data breaches and privacy violations [88], and allows for collaborative model training without the need to transfer raw user data to a central server, ensuring that sensitive information remains on the user's device [89]. Even in this case, model parameters are shared, which can potentially expose user privacy since malicious nodes may try to elicit user preferences from the exchanged model gradients. Another risk can arise from malicious users that perform poisoning attacks by sharing fake gradients to affect the centralized or neighboring node models [90], [91], [92]. Although the decentralized architecture of

FedRS may limit the impact of such attacks [93], [94], the risk still exists, and several techniques have been proposed to tackle this issue. The techniques used for securing user privacy and model robustness to attacks in FedRS follow.

1) PRIVACY PRESERVING TECHNIQUES

The concept of FL involves the participation of multiple parties, including edge devices (such as smartphones or IoT devices) and a central aggregation server [89]. Instead of aggregating data on a single server, the training process is performed locally on each participating device [95]. The models trained on these devices are then shared and aggregated to create a global model [95]. This decentralized approach ensures that user data remains under the control of the individual users, reducing the risk of unauthorized access or data leakage [89].

Obfuscation and noise adding are popular techniques for protecting privacy in collaborative systems. The introduction of *pseudo-items* is a solution aimed at preventing servers from inferring a user's preferences based on a set of items. In the case of FedRS, it involves users uploading gradients not only for items with which they have genuinely interacted but also for a selection of non-interacted, sampled items. Li et al. [72] introduced the FedRec framework, an effective hybrid strategy that generates virtual ratings for items never interacted with by the user. However, a notable limitation of FedRec is its potential to negatively impact model performance due to the introduction of noise. To address this issue, Feng et al. [96] developed an enhanced version, FedRec++, incorporating denoising techniques at the client level. Despite these advancements, the pseudo-items technique, which focuses solely on user interactions without modifying the item gradients, allows a server to potentially infer user-item ratings, as discussed by Chai et al. [67].

Another approach is using encryption techniques, such as *Secure Multi-Party Computation*, to perform computations on encrypted data without revealing the underlying information [97]. *Homomorphic Encryption* is another method to encrypt intermediate parameters before they are uploaded to the server, enabling mathematical operations on the encrypted data, a crucial feature for FedRS [98]. Chai et al. [67] introduced a framework, FedMF, which utilizes Paillier homomorphic encryption to encrypt the gradients of the item embedding matrix, achieving performance comparable to traditional matrix factorization. While this method allows operations on ciphertext, it incurs significant computational overhead and operates under the assumption that there is no secret key leakage from clients. Zhang et al. [99] proposed CLFM-VFL, employing homomorphic encryption to conceal user vectors and clustering to enhance recommendation accuracy and manage matrix dimensions. Additionally, homomorphic encryption has been adapted to utilize personal user information in a privacy-preserving manner, further improving recommendation accuracy [77], [100]. Wu et al. [77] explored expanding the local user-item graph through anonymized neighbors using homomorphic encryption. Similarly, Perifanis et al. [101] implemented the Cheon-Kim-Kim Song (CKKS) fully homomorphic encryption scheme to manage parameters between users' friends post-global model generation. While Homomorphic Encryption effectively protects user ratings and maintains accuracy, its significant drawbacks include high computational costs and the persistent risk of secret key leakage.

The *Secret Sharing* mechanism offers an alternative encryption approach by fragmenting intermediate parameters into multiple segments and distributing them among participants. This approach ensures that the complete reconstruction of the parameters is only possible when all segments are gathered. A practical application of this concept is evident in the ShareMF framework [75], a secret sharing-based federated matrix factorization model. In ShareMF, participants distribute the item matrix gradients among themselves, keeping one segment for them while sharing the others with selected participants. Aggregating these hybrid gradients on the server protects user ratings and interaction patterns from server inference. However, it does not completely eliminate the risk of rated items being leaked to participants who receive the split numbers. To address this vulnerability, Lin et al. [102] introduced a combination of secret sharing and pseudo items mechanisms, enhancing privacy protections. Although FedRS based on secret sharing mechanisms excel in protecting user ratings and sustaining recommendation accuracy with lower computational demands than homomorphic encryption-based systems, they significantly escalate communication costs due to the intricate exchange of segments among participants.

To ensure privacy preservation in FedRS, various techniques and protocols have been proposed. One such technique is *Differential Privacy*, which provides privacy guarantees by adding noise to the model updates before aggregation [97]. This helps protect the privacy of individual users and prevents the inference of sensitive information from the model's weights or parameters [103]. *Local Differential Privacy* (LDP) is a perturbation-based mechanism, which tries to tackle the computational and communication overheads of the encryption mechanisms. LDP uses statistical computations while preserving the privacy of individual participants [51], making it a viable option for perturbing intermediate parameters in large-scale FedRS, especially in industrial applications. Dolui et al. [104], used differential privacy on the item embedding matrix before its transmission to the server for aggregation. However, this method still leaves room for the server to extract information from user-rated items by analyzing changes in the item embedding matrix. To enhance privacy during the model training process, Wu et al. [77] have integrated pseudo items with LDP mechanisms in FedGNN, addressing both user interaction behaviors and ratings. This approach involves clients sampling non-interacted items and creating virtual gradients for these items, alongside applying LDP to safeguard user ratings. This is achieved by clipping gradients and adding Laplacian noise, thus maintaining privacy. Nevertheless, Liu et al. [105] highlighted a challenge

TABLE 4. Summary of Privacy Preserving Techniques

Techniques	Descriptions	Citations
Obfuscation and Noise Adding	Techniques for protecting privacy in collaborative systems.	[72], [96]
Secure Multi-Party Computation	Encrypts data for computations without revealing underlying information.	[97]
Homomorphic Encryption	Encrypts intermediate parameters before upload; allows operations on encrypted data.	[67], [106], [77], [99], [100], [101]
Secret Sharing	Fragments parameters into segments distributed among participants.	[75], [102]
Differential Privacy	Adds noise to model updates before aggregation; protects the privacy of individual users.	[97]
Local Differential Privacy	Uses statistical computations to preserve individual privacy; minimizes computational and communication overhead.	[51], [104], [77], [105]

in this approach: the varying gradient magnitudes during training, suggesting the incorporation of dynamic noise in proportion to the gradient magnitudes. Overall, LDP in FedRS minimizes the computational and communication overhead but introduces noise and has a negative impact on the recommendation model performance.

The different approaches that attempt to preserve user privacy in FedRS are summarized in Table 4.

2) ROBUSTNESS AGAINST MODEL POISONING ATTACKS

The Model Poisoning Attacks can be categorized into two major groups, namely Target Poisoning Attacks and Untarget Poisoning Attacks.

Target poisoning attacks in FedRS primarily aim to manipulate the exposure of certain items, often driven by financial reasons. Zhang et al. [93] introduced a method known as *PipAttack*, which leverages popularity bias to promote specific items in FedRS. This attack aligns targeted items with popular items in the embedding space to artificially boost their rank score. PipAttack is designed to minimize detection and the negative impact on overall recommendation accuracy by maintaining modified gradients uploaded by malicious clients close to normal ones.

Rong [107] proposed *FedRecAttack*, a model poisoning strategy against FedRS that aims to mitigate the recommendation accuracy degradation typically caused by targeted poisoning attacks. It requires fewer malicious clients to be effective. FedRecAttack utilizes a small subset of public interactions to approximate the user feature matrix, which is then employed to generate poisoned gradients.

Both PipAttack and FedRecAttack depend on certain prior knowledge. PipAttack requires access to popularity information, and FedRecAttack needs public interaction data. Therefore, the effectiveness of these attacks is significantly diminished in the absence of such knowledge, making them not universally applicable in all FedRS scenarios.

To address this limitation, Rong et al. [108] developed two methods for generating poisoned gradients without prior knowledge: *random approximation* (A-ra) and *hard user mining* (A-hum). A-ra employs a Gaussian distribution to approximate normal users' embedding vectors, while A-hum optimizes these vectors using gradient descent to identify "hard users". This approach enables A-hum to effectively compromise FedRS even with a minimal presence of malicious users.

The *untargeted attacks* in FedRS are attacks, often orchestrated by competing entities, which do not target specific items, but rather aim to lower the overall efficiency of the recommendation model. A notable example is *FedAttack* [94], which disrupts the training process of FedRS. It employs a globally hard sampling technique [109] to manipulate model training, which involves malicious clients inferring users' interests from local profiles and then deliberately choosing items that align with these interests as negative samples, while selecting completely misaligned items as positive samples. What makes FedAttack particularly effective is its subtlety: it only tweaks training samples and the malicious clients mimic normal users with varying interests. As a result, FedAttack can significantly impair FedRS performance, even when defensive measures are in place.

Several methods have been proposed in the literature for defending against poisoning attacks in FedRS. These methods can be broadly classified into Robust aggregation and Anomaly detection methods.

Robust aggregation in FedRS aims to ensure global model convergence even with up to 50% malicious participants [110]. Various methods have been developed for this purpose. The Median method [110] selects the median value of each model parameter, offering a central representation of the distribution. The Trimmed-Mean approach [110] enhances robustness by removing extreme values from each parameter before calculating the mean, effectively reducing outlier impact. Krum and Multi-Krum [54] strategies select local models close to others for the global model, limiting the effect of malicious inputs. Bulyan [111] combines the principles of Krum and Trimmed-Mean, ensuring model convergence even under complex conditions. Norm-Bounding [112] employs parameter clipping to a fixed threshold for aggregation, mitigating the influence of poisoned parameters. Finally, A-FRS [17] uses gradient-based Krum in momentum-based FedRS to filter out malicious clients. While these strategies do provide convergence guarantees, many, such as Bulyan, Krum, Median, and Trimmed-Mean, can significantly degrade FedRS performance. Furthermore, novel attacks like PipAttack and FedAttack [93], [94] pose additional challenges by closely approximating normal user patterns, complicating defense mechanisms.

Anomaly detection in FedRS is crucial for identifying and filtering out poisoned model parameters from malicious

TABLE 5. Summary of Robustness Against Model Poisoning Attacks

Techniques	Descriptions	Citations
Target Poisoning Attacks	Manipulates exposure of certain items, often for financial reasons.	[93], [107], [108]
Untarget Poisoning Attacks	Aims to lower the overall efficiency of the recommendation model, not targeting specific items.	[94]
Robust Aggregation	Ensures global model convergence; includes Median method, Trimmed-Mean, Krum, Multi-Krum, Bulyan, Norm-Bounding, A-FRS.	[54], [110], [111], [112], [17], [93], [94]
Anomaly Detection	Identifies and filters out poisoned model parameters; includes FSAD and gradient-based features for detection.	[107], [113]

clients during global model aggregation. A notable example is the work by Jiang et al. [113], who developed an anomaly detection strategy called the Federated Shilling Attack Detector (FSAD) specifically for federated collaborative filtering scenarios. FSAD operates by extracting four unique features from the gradients uploaded by clients and then employs these gradient-based features to train a semi-supervised Bayes classifier. This classifier is designed to pinpoint and eliminate poisoned gradients. However, the challenge in FedRS is the wide variation in user interests, leading to a diverse range of parameters being uploaded. This diversity significantly complicates the task of anomaly detection [107].

The different approaches for securing FedRS models against poisoning attempts are summarized in Table 5.

VIII. APPLICATIONS AND EXPECTED BENEFITS

A. APPLICATIONS

The utilization of FedRS offers substantial benefits across various domains, effectively addressing traditional challenges associated with data availability, regulatory compliance, and privacy concerns.

In the realms of *online services* and *advertising*, FedRS can mitigate risks of personal information leakage, which historically has been a concern with centrally stored data. These systems, leveraging Click-Through Rates (CTR) as a basis for user interaction analysis, can enhance both performance and safety. Tan et al. [14] introduced a FedRS model that integrates popular RS algorithms to support diverse online services, aiming for practical real-world application. They also demonstrated a use of their FedRS framework in content recommendation. Additionally, Wu et al. [70] developed a CTR prediction method that utilizes multi-platform user behavior data to discern user interests while circumventing the need for central data storage.

In *healthcare*, FedRS enables the amalgamation of patient information across multiple facilities, fostering the development of personalized models for each patient. This approach ensures the confidentiality of sensitive health information by utilizing a cross-data silo strategy. Song et al. [114] employed the Federated AI Technology Enabler (FABLE) in conjunction with a federated healthcare recommendation platform to generate improved recommendations based on shared user data. A Federated Drug RS, namely F-HRS, has been showcased in [115] in an attempt to provide valuable recommendations without violating the medical data confidentiality rules. The authors adopt a client-server architecture and combine a cross-silo approach, in which each hospital keeps its own data, with a centrally accessible database of drugs (items). They employ the Neural Collaborative Filtering algorithm [100] that is trained using positive and negative interaction examples, and the FedAvg technique for model aggregation at the server. BVFLEMR [116] is another FedRS that employs blockchain technology to guarantee the security of the RS and deliver personalized treatment recommendations without harming patient privacy. A client-server architecture has been selected to support the cross-silo design, in which multiple hospitals store patient records in the blockchain and the central server then uses the data to train the recommendation model. The approach does not adopt any model partitioning schemes, that are common in FL, but is interesting since it focuses on the security aspects of data sharing.

Education is another area where RSs thrive, especially with the recommendation of multimedia educational content in Massive Open Online Courses (MOOCs). To support students with the appropriate recommendations and at the same time protect their privacy, FedRS frameworks like FedMCsRse [117]. FedMCsRse considers the cross-device nature of RSs in MOOCs and proposes a hierarchical reinforcement learning approach in the module recommendation task. It adopts the client-server model, in which selected clients contribute to the updates of the model in each round.

In the field of *energy saving* RSs have been employed in a multitude of ways to support users and improve their energy footprint. The EM^3 project [118] used FL for handling Big Data that come from energy meters, and focused on using edge devices (clients) to learn and update local models, before sending them to the cloud (server) for aggregation. The approach has been experimentally demonstrated in an office setup and demonstrated an increased user acceptance of recommendations, which denoted a high quality of recommendations.

Finally, in the area of *tourism*, FedRS proves its efficacy by integrating user habits and purchase behaviors from diverse platforms to recommend Points of Interest (POIs) without exposing user privacy. The RS can securely extract sensitive check-in information, enabling service providers to offer precise, personalized services and recommendations while maintaining user privacy [7]. A FedRS application in tourism has been proposed in [119], where authors adopted the client-server architecture for model exchange and introduced: i) FCF-CAPR that is based on collaborative filtering and the use of explicit and implicit data feedback, and FedCorr, a

user-attraction data correction model that exchanges only user update gradients with the cloud, which is used to train the model but also to update the POI (i.e., item) information. As a result, the users get better recommendations and at the same time keep their privacy by not exposing their actual activity data.

What is interesting in all the aforementioned applications and application areas is that the developed systems are still in the design and prototype phase, and refer to research prototypes mostly, thus leaving space for larger-scale implementations.

In the following, we provide a sample case of how the online retail industry can benefit from FedRSs.

B. CASE STUDY: ENHANCING PRIVACY IN ONLINE RETAIL RECOMMENDER SYSTEM THROUGH FEDERATED LEARNING

Background and Motivation One of the world's leading online retailers, aims to revolutionize its recommender system by integrating Federated Learning (FL). This case study explores the deployment of FL to enhance user privacy while maintaining or surpassing the current level of recommendation accuracy. The retailer faces challenges related to privacy concerns and the need for more fine-grained personalized recommendations. The conventional centralized recommender system, while effective, raises user privacy issues. The retailer seeks to address these concerns and pioneer a privacy-centric approach through FL implementation.

Privacy Concerns in the Current Recommender System: The retailer's existing RS, while effective in providing personalized recommendations, raises significant privacy concerns. The centralized nature of the system involves the aggregation and analysis of vast amounts of user data, including purchase history, browsing behavior, and personal preferences. Users express apprehensions about the potential misuse or mishandling of their sensitive information, leading to concerns about data security and unauthorized access. The current system's reliance on centralized data processing also means that user data is stored and processed on the retailer's servers. This centralized approach can be a potential target for security breaches, raising additional privacy issues. As users become increasingly aware of the value of their personal data, addressing these privacy concerns becomes crucial for maintaining user trust and satisfaction.

Balancing Privacy Concerns with the Desire for Personalized Recommendations: Despite these privacy concerns, users inherently desire and appreciate the benefits of personalized recommendations. The online retailer's customers are accustomed to the convenience and efficiency that come with receiving tailored product suggestions based on their preferences and past interactions. The challenge lies in striking a delicate balance between addressing privacy concerns and providing users with the personalized experiences they value. Users are often caught in a dilemma where they want to benefit from the advantages of personalized recommendations but are wary of potential privacy infringements. This dichotomy underscores the need for innovative solutions that can reconcile these opposing interests. Federated Learning emerges as a promising approach in this context, offering a way to enhance the personalization capabilities of the recommender system while addressing users' privacy concerns by keeping their data decentralized and secure on their devices. The integration of FL aims to provide users with the best of both worlds - personalized recommendations and robust privacy protection.

Implementation of Federated Learning: The implementation of federated learning on the retailer's recommender system impacts Data Distribution, Model Training, and Model Updates. More specifically, concerning *data distribution* all user data, including purchase history, preferences, and browsing behavior, is distributed across the retailer's vast user base on various devices, such as smartphones, tablets, and computers. During *model training* a global recommendation model is sent to each device and local models are trained on individual devices using their respective data, preserving the decentralized nature of FL. Finally, on the *model update* step only the local model gradients are sent back to the retailer's central server, ensuring that sensitive user data never leaves the user's device.

Benefits and Implications: The benefits of integrating Federated Learning into the online retailer's recommender system include privacy preservation, enhanced personalization, and reduced data transfer. Sensitive information never leaves the user's device and the users retain full control over their data, thus guaranteeing *privacy preservation* and fostering trust. FL enables the recommender system to learn from diverse user behaviors across different devices, leading to more accurate and *enhanced personalized recommendations*. Finally, the system *minimizes data transfer* by transmitting only the model updates, thus resulting in improved efficiency and lower bandwidth requirements.

This short online retailer's case study showcases the need for integration of Federated Learning into recommender systems, emphasizing the delicate balance achieved between enhanced user privacy and improved recommendation accuracy. The example provides valuable insights into the potential of FL to transform the online retailer's recommender system, setting the stage for pioneering advancements in privacy-centric personalized shopping experiences.

C. POTENTIAL BENEFITS OF COMBINING FEDERATED LEARNING WITH RECOMMENDER SYSTEMS

FedRS stand at the forefront of innovation, merging the power of FL with RSs to yield a host of promising benefits. One of the most significant advantages relates to the enhanced privacy they offer, based on the nature of the federated machine learning mechanisms within RSs. By allowing local model training on individual devices and transmitting only encrypted model updates to a central server or neighboring devices, sensitive user data remains safeguarded throughout the learning process. This approach not only prioritizes data protection but also ensures that personal information stays

within users' control. Moreover, the fusion of federated learning and RSs holds the promise of bolstering performance metrics. The collaborative intelligence gleaned from diverse user behaviors across various devices can potentially enhance recommendation accuracy and fine-tune personalization algorithms. This amalgamation paves the way for more refined, tailored recommendations while maintaining a robust shield around user data privacy and security.

IX. CHALLENGES AND LIMITATIONS

Although the decentralized FL approach is widely regarded as privacy-preserving and safe to manipulation attacks, since it does not expose the full knowledge of the recommender and the entire dataset to the end-users, there are still aspects that can be compromised [93]. This section provides an overview of the security challenges associated with FL-based RSs and discusses the potential risks and consequences of data breaches. Consequently, it performs an overview of potential mitigation measures that can protect Federated RSs from security vulnerabilities and threats in the communication channels and computation processes.

A. SECURITY CONCERNS SPECIFIC TO FEDERATED LEARNING IN RECOMMENDER SYSTEMS

One of the primary security challenges in FL-based RSs is the potential for data leakage or inference attacks [120]. While FL aims to keep user data decentralized and encrypted, there is still a risk of sensitive information being inferred or reconstructed from the model updates or gradients transmitted during the training process [120]. These attacks can compromise user privacy and expose personal information or user preferences.

Data breaches in FL-based RSs can have severe consequences. The loss of sensitive user data can lead to privacy violations and the unauthorized disclosure of personal information [121]. Additionally, the exposure of proprietary algorithms used in the RS can result in intellectual property theft and the loss of competitive advantage for the platform or organization [121]. These risks can have significant financial and reputation-related implications.

There are several security vulnerabilities and threats specific to FL-based RSs. Attacks on the communication channels between the participating devices and the central server can compromise the confidentiality and integrity of the transmitted data [122]. Adversaries may attempt to intercept or manipulate the communication to gain unauthorized access to user data or inject malicious updates into the training process [122]. Ensuring secure and encrypted communication protocols is crucial to mitigate these risks.

The computation processes in FL can also be the target of manipulation attacks. Malicious participants or compromised devices can introduce biased or malicious updates to manipulate the training process or compromise the integrity of the global model [122]. The aggregation process itself can be targeted, with adversaries attempting to infer sensitive information from the aggregated model or exploit vulnerabilities in the aggregation algorithms [122]. Robust mechanisms for participant authentication, model verification, and secure aggregation are essential to address these threats.

Another challenge for privacy-preserving RSs that rely on FL is the lack of negative examples that may arise from the non-IID distribution of training samples to the different nodes. Since data are generated locally, the negative feedback for several items may be missing, leading to a significant degradation in the federated RS performance. To address this issue, a novel method has been proposed that aims to address the lack of negative examples in the nodes, by introducing batch-insensitive losses to alleviate the effect of the under-representation of negative samples [123].

The heterogeneity of devices and data sources can introduce variations in the quality and reliability of the local models [96]. Ensuring fairness and robustness in the aggregation process is crucial to prevent bias and maintain the integrity of the global model [89]. Additionally, the configuration and parameter settings of FLSs need to be carefully designed to ensure privacy and security guarantees [97].

B. MITIGATING SECURITY RISKS IN FEDERATED LEARNING-BASED RECOMMENDER SYSTEMS

Mitigating security risks in FL-based RSs requires the implementation of best practices and strategies to ensure the confidentiality, integrity, and privacy of user data. This subsection discusses these practices and strategies, including the use of secure communication protocols and encryption techniques. Additionally, the importance of data ethics guidelines in ensuring the trust and safety of users is introduced, emphasizing the need for transparency and accountability in data handling and algorithmic decision-making.

To mitigate these security concerns, various solutions and techniques have been proposed. Differential privacy mechanisms can be employed to add noise to the model updates, protecting individual user privacy and preventing inference attacks [120]. Secure aggregation protocols based on cryptographic schemes can ensure the confidentiality and integrity of the aggregated model without revealing individual contributions. Additionally, adopting secure communication protocols, such as encrypted channels and secure authentication mechanisms, can enhance the security of the communication channels [122].

Moreover, the use of secure communication protocols is crucial. Secure Socket Layer (SSL) or Transport Layer Security (TLS) protocols can be employed to encrypt the communication channels between the participating devices and the central server, ensuring the confidentiality and integrity of the transmitted data [124]. Additionally, adopting authentication mechanisms, such as digital certificates or secure tokens, can verify the identity of the participating devices and prevent unauthorized access [124].

Encryption techniques are crucial in protecting user data in FL-based RSs [124]. Homomorphic encryption, for example, allows for computations to be performed on encrypted data

without decrypting it, preserving the privacy of user information [125]. This technique ensures that sensitive data remains encrypted throughout the training process, minimizing the risk of data leakage or inference attacks [125]. Improved encryption algorithms, such as the BFV-based homomorphic encryption, can enhance the efficiency of training while maintaining the system's security [125].

In addition to technical measures, implementing data ethics guidelines is essential to ensure the trust and safety of users in FL-based RSs. Transparency and accountability are key principles in data handling and algorithmic decision-making. Organizations should provide clear and accessible information about the data collection and usage practices, as well as the algorithms and models employed in the RS [126]. This transparency allows users to make informed decisions and understand how their data is being utilized.

Accountability in algorithmic decision-making involves establishing mechanisms for auditing and evaluating the fairness and bias of the RS. Regular assessments of the system's performance and impact on users can help identify and address any potential biases or discriminatory outcomes. Additionally, involving a critical audience, such as independent auditors or external experts, can provide an external perspective and ensure the system's accountability [126].

Data quality and integrity are also crucial considerations in mitigating security risks in FL-based RSs [126]. Safeguarding the entire data value chain, from data collection to preprocessing and model training, is essential to ensure the reliability and accuracy of the system. Additionally, Regular data quality checks, data anonymization techniques, and adherence to data protection regulations can help maintain the integrity of the data and protect user privacy [126].

Existing approaches in mitigating security risks in FL-based RSs have certain limitations. One limitation is the dependence of the training algorithm on the specific machine learning objective being pursued. Different algorithms, such as trees, linear regression, logistic regression, and neural networks, have been proposed, but the choice of algorithm may not cover all possible scenarios and may not be optimal for certain types of data or applications [39]. Additionally, while FL aims to keep training data decentralized, there are still potential security risks and vulnerabilities present. Despite the absence of data sharing among participants, security risks can arise from various aspects, highlighting the need for security measures [127].

X. DISCUSSION

Following the analysis performed in the related literature, we can now answer the research questions of Section I.

RQ1: What are the main approaches that allow the use of FL in the recommendation task?

Answer: Based on the analysis performed in Section VII and more specifically in Section VII-A FL can be combined with typical RS approaches such as matrix factorization, deep learning, or meta-learning, to provide a more privacy-preserving solution in which user preferences are used locally for training individual models. The updates of such models are shared with a central node to be aggregated in a global model or with local neighbors to increase the locally aggregated knowledge without exposing the actual user data.

RQ2: Is FL capable of answering the security and privacy concerns of RSs?

Answer: As explained in Section VII-B, FedRS is by design privacy-preserving. However, the privacy and robustness of FedRS solutions can be further improved using obfuscation, noise insertion and encryption.

RQ3: What are the open challenges for the use of FL in modern RSs

Answer: Despite the multiple applications of FedRS and the potential benefits from their usage in terms of privacy protection and robustness, there are still some open challenges, described in Section IX. Despite the attempts to mitigate the various security risks, there is still space for research, especially focusing on the heterogeneity of nodes and their models and parameters, on the encryption of the exchanged information and the robustness of the training mechanism on malicious attacks.

A. SUMMARY OF LESSONS LEARNT

In this survey paper, we have learned three key points to consider in the design and implementation of Federated Learning Recommender Systems (FedRS), namely: evaluation metrics, challenges and limitations, and security considerations. The subsequent paragraphs delve into a more detailed explanation of these lessons.

Evaluation Metrics: Evaluation metrics are imperative in gauging the success of integrating Federated Learning into the recommender system. Precise metrics such as recommendation accuracy, privacy metrics, and efficiency metrics serve as critical benchmarks. *Recommendation Accuracy:* Assessing recommendation accuracy allows us to determine the effectiveness of the new approach in providing personalized suggestions. This involves comparing the performance of the federated recommender system with the traditional centralized data model. *Privacy Metrics:* Privacy metrics help measure the level of user data protection achieved through Federated Learning, ensuring compliance with privacy expectations. These include data leakage and user exposure metrics. Data leakage metrics refer to measures used to evaluate the inadvertent or unauthorized exposure of sensitive information during the processing, storage, or transmission of data. User exposure metrics refer to measures assessing the extent of individual user data visibility during computational processes, particularly in privacy-focused technologies like Federated Learning. *Efficiency Metrics:* Efficiency metrics in the context of recommender systems and Federated Learning (FL) typically involve quantifying the computational and communication efficiency of the system. These metrics assess how well the system performs in terms of resource utilization, such as processing power, bandwidth, and memory. For FL, efficiency metrics may include measuring the reduction in data transfer between local devices and the central server,

evaluating the computational load on the central servers, and assessing the overall scalability of the system. Efficient systems aim to minimize computational costs and data transfer while maintaining or improving recommendation accuracy, ensuring a streamlined and responsive user experience. All the aforementioned metrics can be applied to the methods presented in Table 2.

Challenges and Limitations: Communication overhead and the heterogeneity of local data are some of the key challenges that need to be addressed in the design and implementation of FedRS. *Communication Overhead:* FL introduces communication overhead due to the need to exchange model updates, potentially impacting real-time recommendation responsiveness. Therefore, there is a need to investigate techniques to minimize and optimize communication overhead, ensuring that the FedRS seamlessly scales with ever-expanding user bases. *Heterogeneity of Local Data:* Address challenges arising from the diversity of user behavior across different devices and locations. There is a need for research methods to effectively handle the diversity in local data across different devices and user segments.

Security Considerations: Security is a critical aspect that demands meticulous attention in FedRS. This involves safeguarding the entire FL process to mitigate potential vulnerabilities and protect user data. Key security considerations include secure aggregation and model initialization. *Secure Aggregation Techniques* entails implementing robust methods for aggregating model updates from various local devices without compromising individual user privacy. This ensures that the sensitive information of individual users remains confidential during the aggregation process. *Privacy-Preserving Model Initialization* is concerned with establishing secure mechanisms for initializing the global model without compromising individual user privacy. This is crucial in preventing any unintentional exposure of sensitive information during the start of the federated learning process.

XI. CONCLUSION

In this work, we have systematically addressed the interplay between FL and RSs, with a focus on enhancing privacy and security. Our analysis, grounded in answering the key research questions outlined in Section I, reveals insights and contributions to the field.

Firstly, we identified that FL can be effectively integrated with conventional RS approaches like matrix factorization, deep learning, and meta-learning, as discussed in Section VII and Section VII-A. This integration not only maintains user privacy by localizing preference data but also enhances the overall system robustness through collective model updates. Secondly, our exploration into the privacy and security aspects of FedRS, detailed in Section VII-B, confirms that while inherently privacy-preserving, these systems can benefit from additional safeguards such as obfuscation, noise insertion, and encryption. Lastly, we recognize the ongoing challenges, highlighted in Section IX, particularly in managing the heterogeneity of nodes, securing data exchange, and fortifying the system against malicious attacks.

This research contributes to the evolving landscape of privacy-preserving RSs. By integrating FL into RSs, we found a promising pathway to balancing robust performance with stringent privacy and security requirements. Our findings underscore the crucial role of privacy and security in RSs and illuminate the potential of FL as a transformative solution. As we move forward, stakeholders must continue prioritizing security and privacy in their strategies, ensuring that the delivery of personalized recommendations does not compromise user trust and privacy. The synergy between FL and RSs, as presented in our study, offers a blueprint for a more secure, efficient, and ethically responsible future in data-driven recommendations, marking a significant step forward in the realm of digital privacy and security.

ACKNOWLEDGMENT

The statements made herein are solely the responsibility of the authors.

REFERENCES

[1] Y. Himeur et al., "A survey of recommender systems for energy efficiency in buildings: Principles, challenges and prospects," *Inf. Fusion*, vol. 72, pp. 1–21, 2021.
[2] B. Smith and G. Linden, "Two decades of recommender systems at amazon. com," *IEEE Internet Comput.*, vol. 21, no. 3, pp. 12–18, May/Jun. 2017.
[3] D. Jannach and M. Jugovac, "Measuring the business value of recommender systems," *ACM Trans. Manage. Inf. Syst.*, vol. 10, no. 4, pp. 1–23, 2019.
[4] Y. Himeur, S. S. Sohail, F. Bensaali, A. Amira, and M. Alazab, "Latest trends of security and privacy in recommender systems: A comprehensive review and future perspectives," *Comput. Secur.*, vol. 118, 2022, Art. no. 102746.
[5] Y. Himeur et al., "Blockchain-based recommender systems: Applications, challenges and future opportunities," *Comput. Sci. Rev.*, vol. 43, 2022, Art. no. 100439.
[6] M. Harasic, F.-S. Keese, D. Mattern, and A. Paschke, "Recent advances and future challenges in federated recommender systems," *Int. J. Data Sci. Analytics*, pp. 1–21, 2023.
[7] Z. Sun et al., "A survey on federated recommendation systems," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Feb. 07, 2024, doi: 10.1109/TNNLS.2024.3354924.
[8] L. Yang, B. Tan, V. W. Zheng, K. Chen, and Q. Yang, "Federated recommendation systems," in *Federated Learn.: Privacy and Incentive*, Berlin, Germany: Springer, 2020, pp. 225–239.
[9] M. Asad, S. Shaukat, E. Javanmardi, J. Nakazato, and M. Tsukada, "A comprehensive survey on privacy-preserving techniques in federated recommendation systems," *Appl. Sci.*, vol. 13, no. 10, 2023, Art. no. 6201.
[10] K. Muhammad et al., "FedFast: Going beyond average for faster training of federated recommender systems," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2020, pp. 1234–1242.
[11] A. Jalalirad, M. Scavuzzo, C. Capota, and M. Sprague, "A simple and efficient federated recommender system," in *Proc. IEEE/ACM 6th Int. Conf. Big Data Comput., Appl. Technol.*, 2019, pp. 53–58.
[12] M. Imran, H. Yin, T. Chen, Q. V. H. Nguyen, A. Zhou, and K. Zheng, "ReFRS: Resource-efficient federated recommender system for dynamic and diversified user preferences," *ACM Trans. Inf. Syst.*, vol. 41, no. 3, pp. 1–30, 2023.
[13] V. W. Anelli, Y. Deldjoo, T. Di Noia, A. Ferrara, and F. Narducci, "FedeRank: User controlled feedback with federated recommender systems," in *Proc. Adv. Inf. Retrieval: 43rd Eur. Conf. IR Res.*, 2021, pp. 32–47.

[14] B. Tan, B. Liu, V. Zheng, and Q. Yang, "A federated recommender system for online services," in *Proc. 14th ACM Conf. Recommender Syst.*, 2020, pp. 579–581.

[15] Q. Wang, H. Yin, T. Chen, J. Yu, A. Zhou, and X. Zhang, "Fast-adapting and privacy-preserving federated recommender system," *VLDB J.*, vol. 31, pp. 877–896, 2021.

[16] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith, "Federated learning: Challenges, methods, and future directions," *IEEE Signal Process. Mag.*, vol. 37, no. 3, pp. 50–60, May 2020.

[17] C. Chen, J. Zhang, A. K. Tung, M. Kankanhalli, and G. Chen, "Robust federated recommendation system," 2020, *arXiv:2006.08259*.

[18] S. Zhang, W. Yuan, and H. Yin, "Comprehensive privacy analysis on federated recommender system against attribute inference attacks," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 3, pp. 987–999, Mar. 2024.

[19] D. Roy and M. Dutta, "A systematic review and research perspective on recommender systems," *J. Big Data*, vol. 9, no. 1, 2022, Art. no. 59.

[20] P. Lops, M. De Gemmis, and G. Semeraro, "Content-based recommender systems: State of the art and trends," in *Recommender Syst. Handbook*. Berlin, Germany: Springer, 2011, pp. 73–105.

[21] M. D. Ekstrand et al., "Collaborative filtering recommender systems," *Found. Trends Human–Comput. Interaction*, vol. 4, no. 2, pp. 81–173, 2011.

[22] M. H. Mohamed, M. H. Khafagy, and M. H. Ibrahim, "Recommender systems challenges and solutions survey," in *Proc. IEEE Int. Conf. Innov. Trends Comput. Eng.*, 2019, pp. 149–155.

[23] C. Wang, Y. Zheng, J. Jiang, and K. Ren, "Toward privacy-preserving personalized recommendation services," *Engineering*, vol. 4, no. 1, pp. 21–28, 2018.

[24] Y. Hu, Y. Koren, and C. Volinsky, "Collaborative filtering for implicit feedback datasets," in *Proc. IEEE 8th Int. Conf. Data Mining*, 2008, pp. 263–272.

[25] S. Badsha, X. Yi, and I. Khalil, "A practical privacy-preserving recommender system," *Data Sci. Eng.*, vol. 1, no. 3, pp. 161–177, Sep. 2016.

[26] A. Mehmood, I. Natgunanathan, Y. Xiang, G. Hua, and S. Guo, "Protection of Big Data privacy," *IEEE Access*, vol. 4, pp. 1821–1834, 2016.

[27] M. Hazrati, R. Dara, and J. Kaur, "On-farm data security: Practical recommendations for securing farm data," *Front. Sustain. Food Syst.*, vol. 6, Jun. 2022, Art. no. 884187.

[28] B. Mehta and W. Nejdl, "Attack resistant collaborative filtering," in *Proc. 31st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2008.

[29] W. Zhou, J. Wen, Q. Qu, J. Zeng, and T. Cheng, "Shilling attack detection for recommender systems based on credibility of group users and rating time series," *PLoS One*, vol. 13, no. 5, May 2018. [Online]. Available: https://doi.org/10.1371/journal.pone.0196533

[30] N. Cao, Z. Yang, C. Wang, K. Ren, and W. Lou, "Privacy-preserving query over encrypted graph-structured data in cloud computing," in *Proc. IEEE 31st Int. Conf. Distrib. Comput. Syst.*, 2011, pp. 393–402.

[31] N. Kokciyan and P. Yolum, "PriGuard: A semantic approach to detect privacy violations in online social networks," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 10, pp. 2724–2737, Oct. 2016.

[32] C. Chen, Z. Liu, P. Zhao, J. Zhou, and X. Li, "Privacy preserving point-of-interest recommendation using decentralized matrix factorization," in *Proc. AAAI Conf. Artif. Intell.*, 2018, Art. no. 32.

[33] S. Kim, J. Kim, D. Koo, Y. Kim, H. Yoon, and J. Shin, "Efficient privacy-preserving matrix factorization via fully homomorphic encryption," in *Proc. 11th ACM Asia Conf. Comput. Commun. Secur.*, 2016, pp. 617–628.

[34] Q. Yang, Y. Liu, T. Chen, and Y. Tong, "Federated machine learning: Concept and applications," *ACM Trans. Intell. Syst. Technol.*, vol. 10, no. 2, pp. 1–19, 2019.

[35] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Artif. Intell. Statist.*, 2017, pp. 1273–1282.

[36] K. Cheng et al., "SecureBoost: A lossless federated learning framework," *IEEE Intell. Syst.*, vol. 36, no. 6, pp. 87–98, Nov./Dec. 2021.

[37] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Discov. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, Oct. 2010.

[38] Y. Liu, Y. Kang, C. Xing, T. Chen, and Q. Yang, "A secure federated transfer learning framework," *IEEE Intell. Syst.*, vol. 35, no. 4, pp. 70–82, Jul./Aug. 2020.

[39] P. Kairouz et al., "Advances and open problems in federated learning," *Found. Trends Mach. Learn.*, vol. 14, no. 1–2, pp. 1–210, 2021.

[40] K. Bonawitz et al., "Practical secure aggregation for privacy-preserving machine learning," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2017, pp. 1175–1191.

[41] K. Bonawitz et al., "Towards federated learning at scale: System design," *Proc. Mach. Learn. Syst.*, vol. 1, pp. 374–388, 2019.

[42] Q. Li, Z. Wen, and B. He, "Practical federated gradient boosting decision trees," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 4642–4649.

[43] M. Yurochkin, M. Agarwal, S. Ghosh, K. Greenewald, N. Hoang, and Y. Khazaeni, "Bayesian nonparametric federated learning of neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 7252–7261.

[44] X. Li, K. Huang, W. Yang, S. Wang, and Z. Zhang, "On the convergence of fedavg on non-IID data," in *Proc. 8th Int. Conf. Learn. Representations*, Addis Ababa, Ethiopia, Apr. 26–30, 2020.

[45] H. Wang, M. Yurochkin, Y. Sun, D. Papailiopoulos, and Y. Khazaeni, "Federated learning with matched averaging," in *Proc. 8th Int. Conf. Learn. Representations*, Addis Ababa, Ethiopia, Apr. 26–30, 2020. [Online]. Available: https://openreview.net/forum?id=BkluqlSFDS

[46] H. Kim, J. Park, M. Bennis, and S.-L. Kim, "Blockchained on-device federated learning," *IEEE Commun. Lett.*, vol. 24, no. 6, pp. 1279–1283, Jun. 2020.

[47] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur.*, 2015, pp. 1322–1333.

[48] R. Shokri, M. Stronati, C. Song, and V. Shmatikov, "Membership inference attacks against machine learning models," in *Proc. IEEE Symp. Secur. Privacy*, 2017, pp. 3–18.

[49] 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. Secur. Privacy*, 2019, pp. 739–753.

[50] L. Melis, C. Song, E. De Cristofaro, and V. Shmatikov, "Exploiting unintended feature leakage in collaborative learning," in *Proc. IEEE Symp. Secur. Privacy*, 2019, pp. 691–706.

[51] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proc. Theory Cryptography*, 2006, pp. 265–284.

[52] L. T. Phong, Y. Aono, T. Hayashi, L. Wang, and S. Moriai, "Privacy-preserving deep learning via additively homomorphic encryption," *IEEE Trans. Inf. Forensics Secur.*, vol. 13, no. 5, pp. 1333–1345, May 2018.

[53] S. Hardy et al., "Private federated learning on vertically partitioned data via entity resolution and additively homomorphic encryption," 2017, *arXiv:1711.10677*.

[54] P. Blanchard, E. M. El Mhamdi, R. Guerraoui, and J. Stainer, "Machine learning with adversaries: Byzantine tolerant gradient descent," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, Long Beach, CA, USA, 2017, pp. 118–128.

[55] I. Wagner and D. Eckhoff, "Technical privacy metrics: A systematic survey," *ACM Comput. Surv.*, vol. 51, no. 3, pp. 1–38, 2018.

[56] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Foundations Trends Theor. Comput. Sci.*, vol. 9, no. 3–4, pp. 211–407, 2014.

[57] M. Abadi et al., "Deep learning with differential privacy," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2016, pp. 308–318.

[58] K. Chaudhuri, C. Monteleoni, and A. D. Sarwate, "Differentially private empirical risk minimization," *J. Mach. Learn. Res.*, vol. 12, pp. 1069–1109, 2009.

[59] Q. Li, Z. Wu, Z. Wen, and B. He, "Privacy-preserving gradient boosting decision trees," in *Proc. AAAI Conf. Artif. Intell.*, 2020, pp. 784–791.

[60] G. Andrew, O. Thakkar, B. McMahan, and S. Ramaswamy, "Differentially private learning with adaptive clipping," in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 17455–17466.

[61] S. Song, K. Chaudhuri, and A. D. Sarwate, "Stochastic gradient descent with differentially private updates," in *Proc. IEEE Glob. Conf. Signal Inf. Process.* 2013, pp. 245–248.

[62] M. Sabt, M. Achemlal, and A. Bouabdallah, "Trusted execution environment: What it is, and what it is not," in *Proc. IEEE Trustcom/BigDataSE/ISPA*, 2015, pp. 57–64.

[63] S. Goryczka and L. Xiong, "A comprehensive comparison of multiparty secure additions with differential privacy," *IEEE Trans. Dependable Secure Comput.*, vol. 14, no. 5, pp. 463–477, Sep./Oct. 2015.

[64] G. Kaissis et al., "End-to-end privacy preserving deep learning on multi-institutional medical imaging," *Nature Mach. Intell.*, vol. 3, no. 6, pp. 473–484, 2021.

[65] R. Xu, N. Baracaldo, Y. Zhou, A. Anwar, and H. Ludwig, "HybridAlpha: An efficient approach for privacy-preserving federated learning," in *Proc. 12th ACM Workshop Artif. Intell. Secur.*, 2019, pp. 13–23.

[66] C. Xie, O. Koyejo, and I. Gupta, "Asynchronous federated optimization," 2019, *arXiv:1903.03934*.

[67] D. Chai, L. Wang, K. Chen, and Q. Yang, "Secure federated matrix factorization," *IEEE Intell. Syst.*, vol. 36, no. 5, pp. 11–20, Sep./Oct. 2020.

[68] I. Hegedüs, G. Danner, and M. Jelasity, "Decentralized recommendation based on matrix factorization: A comparison of gossip and federated learning," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discov. Databases*, 2019, pp. 317–332.

[69] C. Chen, L. Li, B. Wu, C. Hong, L. Wang, and J. Zhou, "Secure social recommendation based on secret sharing," in *Eur. Conf. Artif. Intell.* 2020, pp. 506–512.

[70] C. Wu, F. Wu, L. Lyu, Y. Huang, and X. Xie, "FedCTR: Federated native ad CTR prediction with cross-platform user behavior data," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 4, pp. 1–19, 2022.

[71] M. Ammad-Ud-Din et al., "Federated collaborative filtering for privacy-preserving personalized recommendation system," 2019, *arXiv:1901.09888*.

[72] G. Lin, F. Liang, W. Pan, and Z. Ming, "FedRec: Federated recommendation with explicit feedback," *IEEE Intell. Syst.*, vol. 36, no. 5, pp. 21–30, Sep./Oct. 2021.

[73] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, 2009.

[74] A. Flanagan, W. Oyomno, A. Grigorievskiy, K. E. Tan, S. A. Khan, and M. Ammad-Ud-Din, "Federated multi-view matrix factorization for personalized recommendations," in *Proc. Mach. Learn. Knowl. Discov. Databases: Eur. Conf.*, 2021, pp. 324–347.

[75] S. Ying, "Shared MF: A privacy-preserving recommendation system," 2020, *arXiv:2008.07759*.

[76] V. Perifanis and P. S. Efraimidis, "Federated neural collaborative filtering," *Knowl.-Based Syst.*, vol. 242, 2022, Art. no. 108441.

[77] C. Wu, F. Wu, Y. Cao, Y. Huang, and X. Xie, "FedGNn: Federated graph neural network for privacy-preserving recommendation," in *Proc. Int. Workshop Federated Learn. User Privacy Data Confidentiality Conjunction ICML 2021*, 2021.

[78] M. Huang, H. Li, B. Bai, C. Wang, K. Bai, and F. Wang, "A federated multi-view deep learning framework for privacy-preserving recommendations," in *Proc. Int. Workshop Federated Transfer Learn. Data Sparsity Confidentiality Conjunction With Int. Joint Conf. Artif. Intell.*, 2020.

[79] W.-Y. Chen, Y.-C. Liu, Z. Kira, Y.-C. F. Wang, and J.-B. Huang, "A closer look at few-shot classification," in *Proc. Int. Conf. Learn. Representations*, 2019.

[80] A. Nichol, J. Achiam, and J. Schulman, "On first-order meta-learning algorithms," 2018, *arXiv:1803.02999*.

[81] F. Chen, M. Luo, Z. Dong, Z. Li, and X. He, "Federated meta-learning with fast convergence and efficient communication," 2018, *arXiv:1802.07876*.

[82] S. Alfeld, X. Zhu, and P. Barford, "Data poisoning attacks against autoregressive models," in *Proc. AAAI Conf. Artif. Intell.*, 2016.

[83] X. Chen, C. Liu, B. Li, K. Lu, and D. X. Song, "Targeted backdoor attacks on deep learning systems using data poisoning," 2017, *arXiv:1712.05526*.

[84] B. Li, Y. Wang, A. Singh, and Y. Vorobeychik, "Data poisoning attacks on factorization-based collaborative filtering," in *Proc. Adv. Neural Inf. Process. Syst.*, Barcelona, Spain, 2016, pp. 1893–1901.

[85] E. Bagdasaryan, A. Veit, Y. Hua, D. Estrin, and V. Shmatikov, "How to backdoor federated learning," in *Proc. Int. Conf. Artif. Intell. Statist.*, 2020, pp. 2938–2948.

[86] C. Xie, K. Huang, P.-Y. Chen, and B. Li, "DBA: Distributed backdoor attacks against federated learning," in *Proc. Int. Conf. Learn. Representations*, 2020.

[87] L. Su and J. Xu, "Securing distributed gradient descent in high dimensional statistical learning," *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 3, no. 1, pp. 1–41, 2019.

[88] Y. Liu et al., "Boosting Privately: Privacy-Preserving Federated Extreme Boosting for Mobile Crowdsensing," in *Proc. IEEE 40th Int. Conf. Distrib. Comput. Syst.*, 2020, pp. 1–11.

[89] L. U. Khan, W. Saad, Z. Han, E. Hossain, and C. S. Hong, "Federated learning for Internet of Things: Recent advances, taxonomy, and open challenges," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 3, pp. 1759–1799, Thirdquarter 2021.

[90] M. Fang, G. Yang, N. Z. Gong, and J. Liu, "Poisoning attacks to graph-based recommender systems," in *Proc. 34th Annu. Comput. Secur. Appl. Conf.*, 2018, pp. 381–392.

[91] H. Huang, J. Mu, N. Z. Gong, Q. Li, B. Liu, and M. Xu, "Data poisoning attacks to deep learning based recommender systems," in *Proc. 27th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2021, pp. 2154–2164.

[92] H. Zhang, Y. Li, B. Ding, and J. Gao, "Practical data poisoning attack against next-item recommendation," in *Proc. Web Conf.*, 2020, pp. 2458–2464.

[93] S. Zhang, H. Yin, T. Chen, Z. Huang, Q. V. H. Nguyen, and L. Cui, "PipAttack: Poisoning federated recommender systems for manipulating item promotion," in *Proc. Fifteenth ACM Int. Conf. Web Search Data Mining*, 2022, pp. 1415–1423.

[94] C. Wu, F. Wu, T. Qi, Y. Huang, and X. Xie, "FedAttack: Effective and covert poisoning attack on federated recommendation via hard sampling," in *Proc. 28th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2022, pp. 4164–4172.

[95] R. Kerkouche, G. Ács, C. Castelluccia, and P. Genevès, "Privacy-preserving and bandwidth-efficient federated learning," in *Proc. Conf. Health, Inference, Learn.*, 2021, pp. 25–35.

[96] F. Liang, W. Pan, and Z. Ming, "FedRec++ : Lossless federated recommendation with explicit feedback," in *Proc. AAAI Conf. Artif. Intell.*, 2021, pp. 4224–4231.

[97] V. Mugunthan, A. Peraire-Bueno, and L. Kagal, "PrivacyFL," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Magn.*, 2020, pp. 3085–3092.

[98] A. Acar, H. Aksu, A. S. Uluagac, and M. Conti, "A survey on homomorphic encryption schemes: Theory and implementation," *ACM Comput. Surv.*, vol. 51, no. 4, pp. 1–35, 2018.

[99] J. Zhang and Y. Jiang, "A vertical federation recommendation method based on clustering and latent factor model," in *Proc. IEEE Int. Conf. Electron. Inf. Eng. Comput. Sci.*, 2021, pp. 362–366.

[100] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 173–182.

[101] V. Perifanis, G. Drosatos, G. Stamatelatos, and P. S. Efraimidis, "FedPOIRec: Privacy-preserving federated poi recommendation with social influence," *Inf. Sci.*, vol. 623, pp. 767–790, 2023.

[102] Z. Lin, W. Pan, and Z. Ming, "FR-FMSS: Federated recommendation via fake marks and secret sharing," in *Proc. 15th ACM Conf. Recommender Syst.*, 2021, pp. 668–673.

[103] Z. Liu, T. Li, V. Smith, and V. Sekar, "Enhancing the privacy of federated learning with sketching," 2019, *arXiv:1911.01812*.

[104] K. Dolui, I. Cuba Gyllensten, D. Lowet, S. Michiels, H. Hallez, and D. Hughes, "Towards privacy-preserving mobile applications with federated learning: The case of matrix factorization (poster)," in *Proc. 17th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2019, pp. 624–625.

[105] Z. Liu, L. Yang, Z. Fan, H. Peng, and P. S. Yu, "Federated social recommendation with graph neural network," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 4, pp. 1–24, 2022.

[106] P. Paillier, "Public-key cryptosystems based on composite degree residuosity classes," in *Proc. Int. Conf. Theory Appl. Cryptographic Techn.*, 1999, pp. 223–238.
[107] D. Rong, S. Ye, R. Zhao, H. N. Yuen, J. Chen, and Q. He, "Fedrecattack: Model poisoning attack to federated recommendation," in *Proc. IEEE 38th Int. Conf. Data Eng.*, 2022, pp. 2643–2655.
[108] D. Rong, Q. He, and J. Chen, "Poisoning deep learning based recommender model in federated learning scenarios," in *Proc. 31st Int. Joint Conf. Artif. Intell.*, 2022, pp. 2204–2210.
[109] Y. Kalantidis, M. B. Sariyildiz, N. Pion, P. Weinzaepfel, and D. Larlus, "Hard negative mixing for contrastive learning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020, pp. 21798–21809.
[110] L. Lyu et al., "Privacy and robustness in federated learning: Attacks and defenses," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Nov. 10, 2022, doi: 10.1109/TNNLS.2022.3216981.
[111] R. Guerraoui et al., "The hidden vulnerability of distributed learning in byzantium," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 3521–3530.
[112] A. T. Suresh, B. McMahan, P. Kairouz, and Z. Sun, "Can you really backdoor federated learning?," 2019, *arXiv:1911.07963*.
[113] Y. Jiang, Y. Zhou, D. Wu, C. Li, and Y. Wang, "On the detection of shilling attacks in federated collaborative filtering," in *Proc. IEEE Int. Symp. Reliable Distrib. Syst.*, 2020, pp. 185–194.
[114] Y. Song et al., "Federated learning application on telecommunication-joint healthcare recommendation," in *Proc. IEEE 21st Int. Conf. Commun. Technol.*, 2021, pp. 1443–1448.
[115] S. Pinon, S. Jacquet, C. V. Bulcke, E. Chatzopoulos, X. Lessage, and R. Michel, "Federated health recommender system," in *Proc. 16th Int. Joint Conf. Biomed. Eng. Syst. Technol.*, 2023, vol. 5, pp. 439–444.
[116] T. Hai, J. Zhou, S. Srividhya, S. K. Jain, P. Young, and S. Agrawal, "BVFLEMR: An integrated federated learning and blockchain technology for cloud-based medical records recommendation system," *J. Cloud Comput.*, vol. 11, no. 1, 2022, Art. no. 22.
[117] Y. Qin, M. Li, and J. Zhu, "Privacy-preserving federated learning framework in multimedia courses recommendation," *Wireless Netw.*, vol. 29, no. 4, pp. 1535–1544, 2023.
[118] I. Varlamis et al., "Using Big Data and federated learning for generating energy efficiency recommendations," *Int. J. Data Sci. Analytics*, vol. 16, no. 3, pp. 353–369, 2023.
[119] Y. Cai, H. Gao, J. Liao, X. Li, Y. Xu, and J. Xiong, "A personalized recommendation model based on collaborative filtering and federated learning for cultural tourism attractions in Fujian-Taiwan," in *Proc. IEEE Int. Conf. Softw. Syst. Eng.*, 2023, pp. 69–77.
[120] W. Yuan, C. Yang, Q. V. H. Nguyen, L. Cui, T. He, and H. Yin, "Interaction-level membership inference attack against federated recommender systems," in *Proc. ACM Web Conf.*, 2023, pp. 1053–1062.
[121] E. Tankard, D. Hagos, and D. B. Rawat, "Asynchronous federated learning for tactical autonomy," *Proc. SPIE*, vol. 12538, pp. 284–290, Jun. 2023.
[122] M. Mansouri, M. Önen, W. Ben Jaballah, and M. Conti, "SoK: Secure aggregation based on cryptographic schemes for federated learning," *Proc. Priv. Enhancing Technol.*, vol. 2023, no. 1, pp. 140–157, Jan. 2023.
[123] L. Ning, K. Singhal, E. X. Zhou, and S. Prakash, "Learning federated representations and recommendations with limited negatives," in *Proc. Workshop New Front. Federated Learn.: Privacy, Fairness, Robustness, Personalization Data Ownership, NeurIPS*, 2021.
[124] H. Fang and Q. Qian, "Privacy preserving machine learning with homomorphic encryption and federated learning," *Future Internet*, vol. 13, no. 4, Apr. 2021, Art. no. 94.
[125] F. Wibawa, F. O. Catak, S. Sarp, and M. Kuzlu, "BFV-based homomorphic encryption for privacy-preserving CNN models," *Cryptogr.*, vol. 6, no. 3, Jul. 2022, Art. no. 34.
[126] J. Kemper and D. Kolkman, "Transparent to whom? no algorithmic accountability without a critical audience," *Inf. Commun. Soc.*, vol. 22, no. 14, pp. 2081–2096, Dec. 2019.
[127] O. Aouedi, A. Sacco, K. Piamrat, and G. Marchetto, "Handling privacy-sensitive medical data with federated learning: Challenges and future directions," *IEEE J. Biomed. Health Inform.*, vol. 27, no. 2, pp. 790–803, Feb. 2023.

CHRISTOS CHRONIS received the Bachelor of Science (B.Sc.) degree in informatics and telematics from the Harokopio University of Athens, Kallithea, Greece, where he is currently working toward the Ph.D. degree with the Department of Informatics and Telematics. With a primary focus on recommendation systems utilizing reinforcement learning, he strives to optimize personalized user experiences. He has actively contributed to numerous EU projects, showcasing his expertise in machine learning, system architectures, autonomous vehicles, robotics, and the IoT. Through his dedication and passion, he has actively participated in multiple publications featured in international journals and conferences.

IRAKLIS VARLAMIS (Member, IEEE) received the M.Sc. degree in information systems engineering from the University of Manchester Institute of Science and Technology, Manchester, U.K., and the Ph.D. degree from the Athens University of Economics and Business, Athens, Greece. He is currently a Professor of data management with the Department of Informatics and Telematics, Harokopio University of Athens (HUA), Kallithea, Greece. He has more than 250 articles published in international journals and conferences and more than 5000 citations on his work. He holds a patent from the Greek Patent Office for a system that thematically groups web documents using content and links. His research interests include data mining and social network analytics to recommender systems for social media and real-world applications. He is the Scientific Coordinator for HUA in several EU (H2020, ECSEL, REC) and Qatar (QNFR) projects as well as in national projects.

YASSINE HIMEUR (Senior Member, IEEE) received the M.Sc. and Ph.D. degrees in electrical engineering in 2011 and 2015, respectively. Following his doctoral studies, he received the Habilitation to Direct Research, which granted him the official authorization to supervise research, in 2017. He is currently an Assistant Professor of engineering and information technology with the University of Dubai, Dubai, UAE. From 2013 to 2019, he held the position of a Senior Researcher with the Algerian Center for Development of Advanced Technologies, where he was also the Head of the TELECOM Division during 2018–2019. His academic journey led him to join the faculty with the University of Dubai, after serving as a Postdoctoral Research Fellow with Qatar University during 2019–2022. Throughout his career, he has been actively involved in conducting R&D projects and has played a significant role in proposing and co-leading several research proposals under the NPRP grant (QNRF, Qatar). With more than 120 research publications in high-impact venues, he has made valuable contributions to the field. His research interests include Big Data and IoTs, AI/ML/DL, healthcare technologies, digital twins, metaverse, recommender systems, smart grid, building energy management, smart buildings, NLP, generative AI, and cybersecurity. He was the recipient of the Best Paper Award at the 11th IEEE SIGMAP in Austria in 2014.

AYA N. SAYED received the B.Sc. degree in electrical engineering with a minor in computer science and the M.Sc. degree in electrical engineering from Qatar University, Doha, Qatar. She is currently a Research Assistant with the Department of Electrical Engineering, Qatar University. Her research interests include energy efficiency in buildings, machine learning, and Internet of Things applications.

TAMIM M. AL-HASAN was born in Doha, Qatar in 1999. He received the B.S. degree in electrical engineering from Qatar University, Doha, Qatar, in 2022. He is currently a Research Assistant with Qatar University. He has previously worked in both academic and non-academic positions with Qatar University while pursuing the degree. His research interests include embedded systems design, machine learning, computer security, and cryptography. He was the recipient of the several academic and non-academic awards, including the IEEE Regional Exemplary Student Branch Award, when he was an active board member (vice-chair) of the student branch with Qatar University.

ARMSTRONG NHLABATSI received the Ph.D. degree in computer science from The Open University, Milton Keynes, U.K. He is currently a Research Associate with the KINDI Computing Research Centre, Qatar University. He has contributed to security research and innovation in adaptive security for the cloud, threat-specific security risk evaluation, quantification of satisfaction of security requirements, and information security for sports accreditation systems. He is currently working on approaches to analyzing emergent security properties of systems of socio-technical systems and approaches to visualizing security policies. His research interests include security requirements engineering, security risk evaluation, requirements traceability, and the feature interaction problem for information security.

FAYCAL BENSAALI received the Ph.D. degree in electronic and computer engineering from Queen's University, Belfast, U.K., in 2005. During completing his Ph.D., he carried out successful research in the design and implementation of hardware architectures for a range of algorithms for image processing applications. In 2013, he joined Qatar University as an Assistant Professor and is currently a Professor of electrical engineering. He took other academic positions with the University of Hertfordshire, U.K., and Queen's University Belfast, U.K. At the University of Hertfordshire, he was with high-technology companies. He has authored or coauthored more than 200 publications in energy efficiency, machine learning, embedded systems, and reconfigurable computing. He has successfully supervised many Ph.D. students in his research areas which include embedded systems, high-performance reconfigurable computing, image and video processing, machine learning, and energy efficiency. He acted as tutorial co-chair, session chair, and program committee member in many conferences. Prof. Bensaali is a regular reviewer for several international journals and conferences. He is a Senior IEEE Member and an Associate Member of the Higher Education Academy. During his career, he has been awarded several grants from the government and industry.

GEORGE DIMITRAKOPOULOS (Member, IEEE) received the bachelor's degree in electrical and computer engineering from the National Technical University of Athens, Athens, Greece, and the Ph.D. degree from the University of Piraeus, Piraeus, Greece. He has been an Associate Professor with the Department of Informatics and Telematics, Harokopio University of Athens, Athens, since 2010, and with the Department of Research and Development Funding, Infineon Technologies AG (ext.), Augsburg, Germany, since 2018. Since 2002, he has been working in more than 30 EU-funded research and development projects. He is the author of three books and of about 200 publications in international journals and conferences. His research interests include the design and development of strategies for the optimization of vehicular networks based on cognitive networking principles. He is listed in the world's top 2% of scientists according to Stanford University Rankings in 2021 and 2022.